Virus_Checker

COLLABORATORS			
	TITLE : Virus_Checker		
ACTION	*NAME*	*DATE*	*SIGNATURE*
WRITTEN BY		July 29, 2024	

REVISION HISTORY			
NUMBER	DATE	DESCRIPTION	NAME

Contents

Chapter 1

Virus_Checker

1.1 Virus_Checker documentation

```
**************************************************************************

      Virus_Checker Documentation

           by John Veldthuis
          Member of SHI Anti Virus Group

**************************************************************************
```



```
**************************************************************************
```

1.2 Distribution

```
DISTRIBUTION:

Virus_Checker is a freely distributable, copyrighted piece of
software. You do not have to pay money to use it, and may upload it
wherever you choose, but you are not allowed to sell Virus_Checker
for profit, or include Virus_Checker on a disk which is sold for
profit, without the author's (John Veldthuis) permission.  Commodore
have this permission already.

Money is not solicited but would be welcome. I can be contacted at
the address below.
```

Please send me any more new viruses so I can update Virus_Checker,
but please don't send a letter asking for a copy without sending me
money to cover postage and disks. I cannot afford to send everyone a disk
out of my own pocket. If you send just a disk then don't be surprised if
you never see it again.

  John Veldthuis
  21 Ngatai Street
  Manaia, Taranaki
  New Zealand
  Phone +64-6-274-8409

Email addresses:

  FIDO 3:771/440.0
  USENET johnv@tower.actrix.gen.nz

1.3 about Safe Hex International

ABOUT SAFE HEX INTERNATIONAL

SAFE HEX INTERNATIONAL HAS MY PERMISSION TO DISTRIBUTE THIS PROGRAM IN THE
ARCHIVE "THE NEW SUPERKILLERS" OR IN ANY FORM THEY WISH TO

If you know a virus programmer you can get a reward of $ 1000 for
supplying his name and address. The fact is that the law punishes data
crime very severely. (5 years in jail in most countries).

We are an international group with more than 500 members who have started
trying to stop the spread of virus. Let me give you some example:

   1. Our motto is: "Safe Hex", who dares do anything else today?".
   2. A virus bank containing more than 1800 Amiga and PC viruses for
supporting good shareware anti virus programs.
   3. We help people to get money back lost by virus infection.
   4. We write articles about virus problems for about 20 computer
magazines worldwide.
   5. We release the newest and the best virus killers around.
   6. We have more than 35 PC and Amiga "Virus Centers" worldwide
where you can get free virus help by phoning our "Hotline", and
the newest killers translated in your own language at very
little cost.

For more information contact:

   SAFE HEX INTERNATIONAL (Please send 2 "Coupon-Response
   Erik Loevendahl Soerensen   International" and a self addres-
   Snaphanevej 10   sed envelope, if you want infor-
   DK-4720 Praestoe   mation about SHI by letter).
   Denmark
   Phone: + 45 55 99 25 12
   Fax : + 45 55 99 34 98

1.4 Bootblock.Library

BOOTBLOCK.LIBRARY

As from version 6.29, Virus_Checker can use the SHI Bootblock.library by

Johan Eliasson Phone: +46 11 169138
Bäckgatan 6
60358 Norrköping
SWEDEN

What this does is add a brainfile of Bootblock viruses to Virus_Checker. You can update this file as SHI release new brainfiles. This way you can update Virus_Checker to recognize new BootBlock viruses without getting a new program.
The file BootBlock.brainfile has to go in the L: directory and the Bootblock.library file in the LIBS: directory. The Install script will put it there automatically.

1.5 Decrunch.Library

DECRUNCH.LIBRARY

As from version 6.23, Virus_Checker can use decrunch.library by

Georg Hörmann
Am Lahnewiesgraben 19
8100 Garmisch-Partenkirchen
GERMANY

Many thanks for the use of this. The decrunch.library file must be in libs: so that Virus_Checker can use it.

WARNING!WARNING!WARNING

decrunch.library seems to have a problem in that it picks up some files as crunched files when they are not. It then crashes the machine totally. This is not Virus_Checker crashing and I can do nothing about it except to say.

If Virus_Checker crashes when you are scanning file and you have crunched file checking turned on then turn it off and check again. If it still crashes then the problem lies with Virus_Checker. If not then the problem is decrunch.library and I cannot do anything about it.

1.6 Installation

INSTALLATION:

To run Virus_Checker once, either type it's name into a CLI window (while the program is in the current directory), or double-click on it's icon from the Workbench. The program will be active until you quit it or reset

your computer.
Installing Virus_Checker so that it will be active all the while your computer is running is a good idea. This is because viruses can be on any disk you insert into any disk drive. With Virus_Checker always active, you will be protected.

1.7 WorkBench 1.3 install

WorkBench 1.3 install

Under 1.3, to install Virus_Checker so that it will be run whenever you reset your computer, edit your startup-sequence to include simply "Virus_Checker". The program will have to be either in the root directory of the disk you are booting off of, or in the C: directory, for this to work.

1.8 WorkBench 2.xx install

WorkBench 2.xx install

Under the 2.0 operating system, installation is much easier. All you have to do is drag the icon for Virus_Checker into the WBStartup drawer on your Workbench disk (or your boot partition if you use a hard disk), and Virus_Checker will automatically be loaded when the Workbench is loaded.
If you don’t load or use WorkBench then edit the user-startup file in the s: directory and simply include "Virus_Checker" somewhere in it.

1.9 NOTICE FOR USERS OF FILE PACKERS AND CRUNCHERS:

NOTICE FOR USERS OF FILE PACKERS AND CRUNCHERS:

If you use a program such as PowerPacker to make your files smaller then be aware that you must check these files before you crunch them.

```
If the file is infected and you crunch them then VC will not find the
virus in the file unless you have the crunched checking option turned on
```

1.10 Enforcer Users

```
ENFORCER USERS

Just a quick note for people who use the program called enforcer.
Virus_Checker will cause Enforcer hits when it does it's memory scan.
There is no way around this as it is the only way to detect a couple of
viruses.  For each low memory read Virus_Checker will cause 1 enforcer
hits.  This will happen only when Virus_Checker starts up and when you
cause it to do a full memory scan via the menu or 'm' key.  There should be
4 enforcer hits.  The reads are from location $20 and location $6c.
```

1.11 French Users

```
FRENCH USERS

Someone has gone to the trouble of translating the Virus_Checker docs into
French.  If you wish to get them then please read the following

You can get the French doc from AUGL or order it for 20 French Francs at:
        BUGSS c/o Christophe Guillon
        12 allee des ecureuils
        La residence La Chataigneraie Appt 22
        33600 PESSAC
        FRANCE

Direct any questions to rullier@platon.emi.u-bordeaux.fr
                     or 2:324/8.1  2:324/8.3
```

1.12 Command Line Options

```
COMMAND LINE OPTIONS:

The syntax is:

Virus_Checker [-l###] [-t###] [-w###] [-b] [-q] [-i] [-n] [-m]
       -c# [dirname]

Where:

    -l### tells Virus_Checker how far from the left edge of the
          screen to open the Virus_Checker window.

    -t### tells Virus_Checker how far down from the top edge of
          the screen to open the Virus_Checker window.

    -w### tells Virus_Checker how wide you want the window. It has
```

```
          a maximum size of 386 pixels and a minimum of 200.  Any
          numbers out of this range are ignored.

          This is ignored by Workbench 2.0 as there is really no need
          for it due to being able to 'pop' the window up when you
          want it and hide it when you don't want it.

    -b    tells Virus_Checker to send its window to the back of all
          the other open windows.

    -n tells Virus_Checker not to open a window. It will check memory
          and disks inserted but you will have to use the ARexx port
          or the commodities 'Exchange' program (or the hotkey) to
          get it to scan the whole disk for Link/File viruses or to
          view the user interface.  To stop VC, run VC again,  use
          the ARexx port, or send it a Kill command from the
          commodities 'Exchange' program.

    -q tells Virus_Checker to check all memory, files, and disks for
          viruses, then exit.  To check the dh0: partition and exit,
          do the following:  "Virus_Checker -q dh0:".  This will check
          memory, disks, files, and dh0:, then exit.

    -i tells Virus_Checker not to put up a requester when it can't read
          the bootblock of a disk.

    -m tells Virus_Checker to watch the file s:startup-sequence for any
          changes. Some viruses will change this file and VC will
          catch it. (Only works under WB2.0 and above)

    -c# where # is 0 or 1
        If -c0 is used then checking inside crunched files is turned off
        if -c1 is used the the checking is turned on.

    dirname is the directory/file you want checked for File Viruses on
            startup.  An example to open the window at x/y position of
            200/100 and check DH0: is: "Virus_Checker -l200 -t100 dh0:".

Virus_Checker l 10 top 20 b i dh0:test
This will set the VC window at x/y position of 10,20, make it into a
backdrop window, ignore errors from the BootBlock reads and check the file
dh0:test when it starts up.

For the window coordinates, any values outside the size of the WB
screen are ignored and any non numerical values are ignored. There
must be no spaces between the options and the numbers. Options may be
given in any order.

If Virus_Checker is already running, and you invoke it again from the
command line, it will pop open the already running version.
```

1.13 Command Line Options

```
For WB2.0 users the command line is
```

```
L=LEFT T=TOP B=BACKDROPWINDOW N=NOWINDOW Q=QUIT I=IGNOREBB W=WATCHSS
  CHECKCRUNCHEDON CHECKCRUNCHEDOFF K=KILL S=STDOUT IGNORECAPTURE DIR

L=LEFT is tells Virus_Checker how far from the left edge of the screen to
      open the Virus_Checker window.

T=TOP tells Virus_Checker how far down from the top edge of the screen to
     open the Virus_Checker window.

B=BACKDROPWINDOW sets the Virus_Checker window as a Backdrop window

N=NOWINDOW Virus_Checker not to open a window.  It will check memory and
     disks inserted but you will have to use the ARexx port or the
     commodities 'Exchange' program (or the hotkey) to get it to scan the
     whole disk for Link/File viruses or to view the user interface.  To
     stop VC, run VC again, use the ARexx port, or send it a Kill command
     from the commodities 'Exchange' program.

I=IGNOREBB tells Virus_Checker not to put up a requester when it can't read
     the bootblock of a disk.

W=WATCHSS tells Virus_Checker to watch the file s:startup-sequence for any
     changes.  Some viruses will change this file and VC will catch it.

CHECKCRUNCHEDON turns on checking inside crunched files

CHECKCRUNCHEDOFF turns off checking inside crunched files

K=KILL will delete files with LINK viruses in them and not try to remove it

Q=QUIT tells Virus_Checker to check all memory, files, and disks for
     viruses, then exit.  To check the dh0:  partition and exit, do the
     following:  "Virus_Checker QUIT dh0:".  This will check memory, disks,
     files, and dh0:, then exit.

S=STDOUT  This is a special mode for Virus_Checker. It implies that
Virus_Checker will quit as soon as it finishes it's checks, Will not put up
any requesters, opens no window, and if it finds any virii it will not
delete them but will write the name of the file and virus to the shell from
which it started. This can be used by BBS operators to check archives
automatically. a line like
Virus_Checker >ram:infected STDOUT DH0:
would check all files in DH0: and if it found any would write the results
to a file called ram:infected.

IGNORECAPTURE tells Virus_Checker to ignore the initial check on the
capture vectors. It will still warn you of changes while it is running.

BBLIB This tells Virus_Checker to use the SHI BootBlock.library. This
should be used as any viruses detected by this library will not be added to
the normal Virus_Checker checking. You need BootBlock.library in the LIBS:
directory and BootBlock.brainfile in L:

DIR is the name of a dir/file to check for viruses on startup
```

1.14 THE WORKBENCH STARTUP

```
THE WORKBENCH STARTUP:

SPECIAL NOTE:

If Virus_Checker is not run from Workbench it will look for the file
S:VIRUS_CHECKER.INFO  This is just a standard workbench info file and can
be used as described in the next section. This is to allow 1.3 users who
run VC from their startup-sequence to config VC easily. It will work for
2.0 users as well. I have done it this way because it is too hard to find
where a program ran from under 1.3. This way I only have to look for 1 file
in one directory.
To use it add the stuff you want under Workbench and save it. Then copy the
Virus_Checker.info file to the S: directory.

Support for the icon stuff has now been put in. These will override the
default settings and also the settings in the S:Virus_Checker.config file.
It will only affect those things that are given in the ICON. The rest will
be left as default or as the config file sets them.

The things that you can put in via the Information menu on Workbench are as
follows. These will be used if VC is started by Workbench
HOTKEY is only used by WB2.0


HOTKEY=string           /* HOTKEY=lcommand shift del                  */
LEFT=num                /* LEFT=150                                   */
TOP=num                 /* TOP=25                                     */
WINDOW=ON/OFF           /* WINDOW=ON or WINDOW=OFF                    */
RESIDENT=ON/OFF         /* RESIDENT=ON or RESIDENT=OFF                */
IGNOREBBERROR=ON/OFF    /* Ignore BootBlock Read Error                */
                        /* use IGNOREBBERROR=OFF to turn requester off*/
WATCHSS=ON/OFF          /* WATCHSS=ON or WATCHSS=OFF                  */
CHECKCRUNCH=ON/OFF      /* Turn on/off Crunched file checking         */
DF0=ON/OFF              /* DF0=ON or DF0=OFF                          */
 |
 V                ;If Off VC will not check BootBlock or startup-sequence
DF3=ON/OFF
FULLCHECKDF0=ON/OFF     /* FULLCHECKDF0=ON or FULLCHECKDF0=OFF        */
     |
     V               ;If ON VC will scan all files on the inserted disk.
FULLCHECKDF3=ON/OFF
BBLIB=ON    /* Tells VC to use BootBlock.library        */


IN ALL CASES DO NOT USE THE QUOTE MARKS " or ' in any place. VC can see the
spaces between strings without them.
```

1.15 The ARexx Interface

```
THE AREXX INTERFACE:

VC has an ARexx port, which means you can send VC commands using the
```

REXX language, available from your Amiga dealer, or as part of the 2.0
Operating System. The port name is "Virus_Checker". Be aware that
case is important and ARexx will not find it if the name is not spelled
right. Here is an example ARexx program that talks to VC:

```
        /* ARexx programs must start with a comment */

        address 'Virus_Checker'     /* Talk to Virus_Checker                  */
        'checkdrive\df0:'           /* Make virus_Checker check df0:         */
                                    /* for viruses                          */
        'scanforsaddam\df0:'        /* Make VC check df0: for Saddam         */
                                    /* virus damage                        */
        'quit'                            /* Make Virus_Checker shut down.         ↩
           */
        'drive\df1: off'            /* Turn off df1: from being scanned  */
```

Notice the '\' between the command and the drive name in the middle
examples. This must be put between all commands and their options. 'quit'
does not take an option so does not need the '\' character there.
Virus_Checker will take the following commands:

```
        checkdrive\drivename      Check drive 'drivename' for file viruses.
        scanforsaddam\drivename   Check drive (DF0:-DF3) for Saddam damage.
        quit                      Make Virus_Checker shut down.
        saveconfig                Save the s:Virus_Checker.file file
        window\option             Open or Close window (Option = on or off)
        drive\df?: option         Turn on/off Drive scan (Option = on or off)
        resident\option                 Turn on/off Resident flag ""      ""  ""
        checkfile\device:dir/filename
        checkbootblock\df?:       Check the Bootblock in df? for viruses
```

Special note for 'checkfile' command.
This one turns off any requesters while doing it's work. If the command
OPTIONS RESULTS is used it will return RESULT if no virus found or if a
virus is found then the string VIRUSNAME Virus was/is present in the file.
This does not mean the virus is gone as there may have been errors trying
to remove the virus.
This is really for BBS users who want to check files as they come in.
You could write an arexx script to search files and log any that come up
with viruses. Later after findong which ones where infected you would run
VC over them again via the main menu thus making sure they where clear.

CheckBootBlock command
This one also needs the options results and returns messages.
If the disk is clear or you give it a number outside the range of df0: to
df3: it will return 'Okay', if VC had trouble reading the disk the message
returned is 'ERROR reading BOOTBLOCK', if the bootblock is Not the normal
one then 'NON-STANDARD BOOT CODE' is returned. If the Bootblock is
infected then the virus name will be returned. At present there is no way
to clear the virus from Arexx but I am working on it. Requesters are
disabled while this is done.

1.16 Virus_Checker Operation

VIRUS_CHECKER OPERATION:

Upon running Virus_Checker, it will first check your memory for viruses and tell you if any were detected. They will either be removed or disabled. Next all disks in the floppy drives will be checked. Any disk put in any drive (df0: to df3:) will be checked.

If Virus_Checker finds and disables the LAMER virus in memory, the machine may guru. Once the machine is reset, however, the virus should be gone.

1.17 The 2.0 User Interface

THE 2.0 USER INTERFACE:

Many Thanks goes to Steve Tibbett for designing and most of the C code for this section. All I did was translate it into assembly and intergrate it into Virus_Checker.

This section describes the user interface that Virus_Checker uses when Kickstart 2.0 is detected in your computer. This section does not apply for users with Kickstart 1.3.
Kickstart 1.3 users can see the special note for using the Config file below.

Virus_Checker can be used either with a window open, or with no window open. When used with the window closed, Virus_Checker will only show itself when it has something to tell you about. If you insert a disk containing a virus, Virus_Checker will pop up a requester telling you about it, and give you some options to deal with it.

The normal Virus_Checker user interface can present itself in two forms. One is the 'TitleBar Window', where only the close gadget, the depth gadget, the Zoom gadget, and the program name are visible.

If you click the Zoom gadget, Virus_Checker's window will change into a window occupying nearly half a normal 640x200 Workbench screen. This window is broken up into three sections: The Preferences section, the Files section, and the Drives seection.

In the Preferences section, you can tell Virus_Checker whether it should open a window or not, whether the window should be a Backdrop window, and whether Virus_Checker should quit immediately when run, or whether it should stay resident. You can also set the window position, and the hotkey that will call Virus_Checker when you want to open it's window or pop it to the front. (The hotkey format is described in the AmigaDOS 2.0 manual, in the section on the commodities exchange). As from 6.05 you can also tell Virus_Checker to ignore errors when reading the BootBlock. It will be saved in the config file.

THE DEFAULT HOTKEY is Left-Amiga Shift del

The Files section is where you list the drives or directories that

Virus_Checker will check when you click the Check button. If you 'Add' DF0: and DF1: to the list, then choose Check, then Virus_Checker will check all the files on both DF0: and DF1: for file viruses.

The Drives section lets you specify which of your floppy disk drives will automatically be checked for bootblock and file viruses when you insert a disk. If you have a program like CrossDOS and you don't want Virus_Checker looking at the msdos disks then simply disable it and Virus_Checker will never look at that drive again. Unless you enable it again.

The Second row of Drive gadgets turn on and off the automatic scanning of the entire disk. These are disabled by default. If you turn them on, then Virus_Checker will scan the entire disk every time you insert one. Checking for file viruses takes some time, so you may not want this on for a drive that you are constantly moving disks in and out of.
The state of these gadgets is also saved in the Config file.

Any of the Gadgets that have text with an UnderScore beneath then can be accessed by simply pressing the that key on the keyboard.
For example. If you wished to change the Hotkey you will notice that the H in HotKey is Underlined. This means simply by pressing the 'h' key that gadget will become active.

The options that you set in the user interface can be saved to disk using the Save button. The options are saved to the file "S:Virus_Checker.Config", and are read from there whenever the program is loaded.

KEYSTROKES:

The Following keys will activate the following functions, when typed into the Virus_Checker window:

s - Will activate the Scan mode
m - Will immediately do a complete memory scan (same as startup)
f - Will activate the Saddam Disk Scan (used to fix Saddam virus damage)
0 - 3 Will check the First File in startup-sequence and bootblock on disk in drive which matches number

There are also some options on the menu (hold the right mouse button to get to the menus) which have keyboard-equivalent shortcuts. These are next to the inverse A on the menu.

LINK/FILE VIRUS CHECK:

If you want to check a disk for Link/File viruses then put the disk in any drive. Make sure the Virus_Checker window is active and use the right mouse button to bring up the Project Menu. Select the "Link/File Scan" and release the mouse button. An alternative way is to just press the 's' key on the keyboard.

This will bring up a requester asking you which drive to check. Enter the drive name in the box, eg. DF0:, DH1:,RAD: etc. Under WB2.0 you can also use the "Use Requester" option. It will then check all the files on that drive. You can also enter directories if you want to

eg, c: df0:c, df0:libs etc.

When Virus_Checker is scanning the disk and you know that a directory is clear and don't want to check it press control-d in the window with the filenames and Virus_Checker will ignore that directory and go back up one level.

If you want to stop the check completely press control-c in the window with the filenames and Virus_Checker will print a break message then stop scanning the disk and go back to normal scanning.

If Virus_Checker brings a requester up that says a program just run has infected your memory with the Xeno Virus, it has already disabled it. You should immediately check all files on the disks that are in the drives at that time. This means that a program that you just ran or a program some other program just ran is infected with the virus and all files should be checked to find out which one it was.

With viruses which use a RomTag I have decided to clear out all RomTags to make sure I remove the Viruses from the list. In doing this you will lose things like Recoverable ram disks such as RAD:, VD0: etc. If you have a virus make sure that you save anything in the ram disks that you want before rebooting. The ramdisks and others will disappear on a reboot. My policy is better safe than sorry.

BRAINFILE ADDITION:

When VC finds a Non-Standard bootblock it will bring up 4 gadgets. One of these gadgets is Learn. Pressing this will allow VC to remember this BootBlock and not bother you again with it. To do this VC writes a file called VCBrainFile to the S: directory. If you have a single drive this will invoke a requester asking that Volume something be put in the drive. This will then save to the file. On Startup VC will check for the file in the S: directory and read it if it is there. If not it will carry on without it. If you get an error then VC will tell you about it and will happily write over the file next time.

NON-STANDARD BOOT CODE:

When Virus_Checker brings up a Requester that says the disk has non-standard boot code, this means that the code in the boot block is not what should be there. This does not mean that it is a virus as many games use copy protection in their boot blocks, and there are many bootblocks that do interesting things, that are not viruses. You should however be cautious if it is not a game. Do not replace the boot block if you are not sure. If something strange happens then please send a copy of the disk to me so that I can check it out. To determine if an unknown bootblock is likely a virus:

1. Format a blank disk so you know it is clear.
2. Make sure all disks except the one just formatted are write protected.
3. Boot from the disk that you suspect.
4. Place formatted disk in drive zero and then reboot.
5. Take disk out of drive zero and turn off computer for about 30 secs.

6. Run the Virus_Checker program. If the Virus_Checker finds non-standard boot code on the newly formatted disk, you have found a new virus. Please send it to me.

1.18 Credits

CREDITS:

My thanks go out to...

Steve Tibbett. For designing and most of the C code for the 2.0 User Interface on Virus_Checker.

Georg Hörmann. For the inclusion of the DECRUNCH.LIBRARY.

Tim Nugent. For the conversion of this doc file to AmigaGuide format.

ARexx. Developed on an Amiga 1000 and is a 100% Amiga product.

John Veldthuis.

1.19 VIRUSES VIRUS_CHECKER DEALS WITH:

VIRUSES VIRUS_CHECKER DEALS WITH:

Virus_Checker deals with many bootblock viruses, some of which are not listed here. The ones that are listed here describe all the types of bootblock viruses, so listing all the rest of them would be redundant.

SCA:
The SCA is the simplest virus to deal with, as it's not actually DOING anything except hiding in memory, until you reboot. We just look at CoolCapture and fix it to get it out of RAM.

AEK:
This is a clone of the SCA virus and we get rid of it in the same manner.

LSD:
Another SCA clone and uses the same code.

BYTE BANDIT:
The Byte Bandit virus takes the DoIO() vector and re-directs it through itself. Thus, any attempt to read or write the boot block (ie, AmigaDOS trying to figure out what kind of disk it is) results in the BB writing itself onto that disk. We couldn't just rewrite the boot block, we have to get him out of RAM first. This virus also

has an interrupt that crashes the machine every 5 minutes or so after it's infected a few of your disks. Ow. It stays in memory not via the Capture vectors, but by a Resident module. When machine looks crashed press these keys at the same time from left to right LAlt, LAmiga, Space, RAmiga, RAlt. This will restore things for another 5 minutes.

REVENGE:
Basically, a Byte Bandit clone except it will bring up an obscene pointer a few minutes after you reboot. We treat it much like the byte bandit.

BYTE WARRIOR
Jumps right into 1.2 Kickstart. Won't work under 1.3. Hangs around via Resident struct, doesn't do any damage.

NORTH STAR/STARFIRE:
Like SCA, hangs around via CoolCapture, killing CoolCapture kills the North Star.

OBELISK SOFTWORKS CREW:
Hangs around via CoolCapture, also watches reads of DoIO() (but doesn't infect EVERY disk - only ones you boot from).

IRQ:
This is the FIRST Non-Bootblock Virus. It copies itself from place to place via the first executable program found in your startup-sequence. It SetFunction's OldOpenLibrary(), has a KickTagPtr, and lives in the first hunk of an infected program.

PENTAGON CIRCLE:
This one looks at the DoIO vector, and has a CoolCapture vector. It will write itself over any virus inserted, but not onto anything else. No danger, easy to eliminate. Holding left button while booting with this one shows different screen colour, but doesn't get rid of it.

HCS:
Hooks into the System Z protector. This is another virus protector that can write itself to disks. Anything that spreads itself, under any name, is a virus. Doesn't do anything except during a reboot, then examines disks and writes over viruses.

DISK-DOKTORS:
This is another virus which looks at the DoIO routine for the reading of any bootblocks. If it finds one it will rewrite a copy of its code to it if it can. This one also patches into the Vertical Blank interrupt and seems to format your disk after a certain number of interrupts (can't be sure though).The nasty bit is it also creates a task called clipboard.device which spends its life copying itself through memory fragmenting the memory into small blocks. Calls ROM CODE direct so won't work under V1.3. We restore the DoIO routine, the Vert Blank interrupt and RemTask the clipboard.device.

LAMER EXTERMINATOR:
This virus was sent to me by Andrew Mercer of the Palmerston North group. His letter said that He noticed strange things on his disks. On disassembling the virus I found that most of it was encrypted and

the data was encrypted randomly using the beam position of the screen. Thus it appears different each time. It patches the trackdisk.device to look at reads and writes, It patches the Sumkick vector in exec in case someone tries to get rid of it. When it detects a read or a write it will randomly select a sector on the disk and will check if it is a data block. If it is it will write LAMER! all over the sector and rewrite it. Some say this Virus will write to write protected disks. I have not had this happen to me and I can see no special code in the disassembly to accomplish this feat.

TIMEBOMB:

This is a strange Virus. It does not insert itself into any vectors. However it will copy itself back to the disk it came from. When the count gets to 2 it will wipe out the Root Directory of the boot disk and display an alert. If the count is over 2 it will just display an alert.

GADAFFI:

Inserts itself into the CoolCapture vector, Uses a RomTag structure and patches the DoIO vector. Jumps directly into the Kickstart so will only work under V1.2 Kickstart. After 13 copies it will step the heads of drives 0 and 1 in and out. We simply clear all vectors and Use the old V1.2 DoIO code entry point.

BSG9:

This is similar to the IRQ virus in that it does not live in the Boot Block. It operates differently. Inserts itself into the RomTag pointer. It then loads the program it replaced and executes it. On Reboot the RomTag is called. It patches the Intuition OpenWindow Routine to its code. It then returns. Once AmigaDos opens up the CLI window the virus code gets run. This gets the startup-sequence file and gets the first command that is run. It then checks if it is already here. If not, then it moves this program from its directory into the devs: directory and renames it a strange name.It then copies itself to replace the command it just moved. A give away is the file size. The Virus size is 2608 bytes and there will be a file with what looks like spaces for its name in the devs: directory. To get rid of it we copy the file in devs: back to the c: directory and rename it. Then delete the file in the devs: directory. In memory all we do is change the RT_INIT code which is run on reboot to do an immediate RTS. The memory for the program is still used but the Virus is disabled. It will display a screen of its own which says:

```
A Computer Virus is a disease
Terrorism is a Transgression
Software Piracy is a crime
This is the Cure
BSG9 [plus some other junk]
```

WAR HAWK:

This Virus installs itself into the CoolCapture Vector. It copies itself to the disk when the computer is warm booted. After every four copies it displays a message. To get rid of it we simply clear the CoolCapture vector.

VKILL (or AIDS):

This is another virus hidden as a Virus protector. When booted it

copies itself to the stack area that is not used. It then patches the CoolCapture vector to survive a reboot. It patches the PutMsg vector of ExecBase to watch for BootBlock reads and writes.When it finds one it checks it and tells you if a virus is present.If you want to get rid of it it will copy itself to the disk. To remove it we Clear the CoolCapture Vector and SetFunction the PutMsg vector

ULTRAFOX:
This one lives in the CoolCapture vector. When you reboot it will change the DoIO vector and wait for a BootBlock read.When it finds one and the disk is not already infected it will write itself to the bootblock. After every 16 copies it will put a custom copper list which displays greetings.

PVLPROTECTOR:
This one is another bootblock protector. When it finds a virus it will write itself to the disk instead of a proper bootblock. All we do is set the RomTag to do a RTS.

REVENGE LAMER EXTERMINATOR:
This is another file virus. It is supposed to speed up disk operations by 800%. This was found on a BBS and when run patches itself into several places. It will read the s:startup-sequence file on reboot and will edit it so that it runs itself as the program. It sticks out because the first line in the startup-sequence will be blank. When the Checker finds it look in the Root directory and you will find what looks like a blank filename. Virus Checker will rename this virus for you. You can then delete the virus and alter your startup-sequence to get rid of the first blank line

UNKNOWN:
This is a virus that has no names anywhere and will only work under V1.2 Kickstart. Very easy to get rid of.

JITR:
Very mild sort of virus this one. Only writes itself to the BootBlock. Does nothing else. Easily fixed by clearing the CoolCapture vector.

MICROSYSTEMS:
Haven't got this one yet so can't tell you much about it.
Just have to restore a vector in the exec.library and clear the Exec CoolCapture vector.

XENO:
This virus is a very nasty one in the way that it infects all programs that can be run. It does not need the program to be run but even someone doing a LIST or DIR on a disk when the virus is present will infect all those other files on disk. It patches into the dos.library and takes over the Open(), Lock() and LoadSeg() calls in dos. This way it can intercept the files being looked at. It will copy itself to the start of every runnable program and alter the file so that it still works. There is also an encrypted message which says 'Greetings from the Xeno Virus' but I have not worked out when this appears yet. To get rid of it from memory we have to reset the changed vectors. To get rid of it from the file is very much harder.

First the file has to have the virus removed from the code. Then the relocation data pointers have to be changed so that everything still works. When Virus_Checker finds a file infected with the Xeno Virus it will tell you which file it is and bring up a requester. You can now check the files on drive zero for further viruses if you want.

16 BIT CREW:
This virus does not do much and only infects disks that you boot with. To get rid of it from memory we clear the CoolCapture Vector and restore the DoIO vector.

NEW ALIEN BEAT:
This one will only work under Version 1.2 Kickstart as it jumps into the ROM code directly. To fix in memory we have to manually patch the DoIO vector and FindResident Vector with the correct values for 1.2. and clear the Capture vectors.

BLACKFLASH:
This virus will display a message after a certain amount of copies of it have been made. It says that your computer is sick and has a virus. To remove it we just restore the DoIO vector and clear out the capture vectors.

DIGITAL EMOTIONS:
This is another tame virus. Only infects disks when it is rebooted. Clean out the Captures vectors and it is gone.

SCARFACE:
This takes over the BeginIO routine in the trackdisk.device to watch for reads and writes to the disk. When it finds one it will write itself to the disk. It also has a VertBlank interrupt which will do something after a while. I think it only reboots the machine. It also has a romtag which we have to clear out.

TURK:
Another simple virus. Does not do very much. Simple to get rid of.

JOSHUA:
Again, lives in the TrackDisk BeginIO and VertBlank Interrupt. Also has a RomTag to survive reboots. This one will display a sprite after so many interrupts. I am not sure what it looks like but maybe someone wants to wait until it is triggered. It counts interrupts. It will also infect every disk but in the drive that is not write protected. Data in it that says something is encoded. To remove we simply restore the BeginIO code and VertBlank Interrupt and wipe out the RomTag.

BUTONIC:
This is another file type virus. It uses the DoIO vector to check for reads to the Root Block of a disk. It will then write the virus to the disk and add it to the startup-sequence as the first instruction. The filename of the virus and its comment make it invisible when doing a DIR but shows up with a LIST. This will also bring up GURU messages and change the title of the active window to some german stuff. To get rid of it we clear the ROMTAG, restore the DoIO vector and delete the file off the disk. You will need to remove the blank line from the startup-sequence where the

virus was. The second version of this infects the Level 2 Interrupt as well and uses different file names to hide itself in the Startup-Sequence.

CENTURIONS;
Another file type virus. It hooks into the Trackdisk BeginIO() vector and waits for reads to the boot block of a disk. It changes the SumKickData() vector so that it will survive a checksum. To get rid of it in memory we simply kill the RomTag vector, restore the SumKickData vector and patch the trackdisk code it uses to skip over the virus. When it finds a read to the bootblock it will check the write protect. It will then find the startup-sequence and find the name of the first command. It then looks for the command in the root directory, then the c directory. Once found it adds itself to the front of the file and is run when the startup-sequence is run again. Signs of infection are that it adds 3916 bytes to the size of the file it infects. After every ten copies it will change the pointer to a smiley face and a message will scroll across it.

CODERS NIGHTMARE:
A boot block virus. Fairly tame this one but it will wreck copy protected disks. It takes over the DoIO vector waiting for reads to track zero block 0 then it writes itself to the disk if it can. It has a level 2 interrupt which after a time will display a message and then reboot the machine. To remove we just reset the DoIO and Level 2 Interrupt vectors and clear out the RomTag.

FORPIB:
Another boot block virus. It takes over the Trackdisk BeginIO vector and waits for reads to block 0. Then it copies itself if it can. It also has a VertBlank Interrupt and after a certain time a message will appear. (I think). There is a bug in this in that it tries to use a color register but it has got the wrong value in there. To remove just restore both vectors and remove the RomTag.

GX TEAM:
Yet another bootblock virus. This just takes over the DoIO vector and after a certain number of copies it will bring up a requester then guru. To remove replace the DoIO vector and clear RomTag and Capture vectors. This virus will only work under version 1.2 kickstart.

GREMLINS:
Yes, another bootblock virus. Sickening isn't it. Don't know what this one does but very easy to remove. Just zero the Capture vectors, restore the SumKickData vector and DoIO vector and it's gone.

KAUKI:
This boot block virus will only work under Version 1.2 kickstart. As I don't have it I can't tell you what is displayed but something is displayed. Easy to get rid of. Just clear the Capture vector and set the DoIO vector to $FC06DC just to make sure.

SADDAM virus
This is a file type file that hides itself as the Disk-Validator. The disk on which it came was unvalidated so AmigaDOS loaded it to try and validate the disk. This causes the virus to run and

infect your machine. It does infect a lot of vectors that need fixing when it is found. I just wipe it off the disk and it is left to the user to put a new Disk-Validator on the disk.

It will change the root block BitMap pointer so that if the virus is not running AmigaDOS will think the disk is UnValidated and load the virus. It will also change DATA blocks so DOS does not know them unless the virus is running. When the virus is triggered it will wipe out the whole disk and bring up a Requester telling you it is the SADDAM virus.

CCCP:

This a combination Bootblock and file virus. It changes itself so that it will write to the BootBlock and to random files on the disk. The only way to find it on disk is to scan the whole disk.

DISASTER MASTER 2:

This is a fairly simple File type virus. It will write to a disk after a warm boot and if there is enough room on the disk will make a file called cls in the C: directory and add cls * as the first line in the startup-sequence. We just clear the RomTag and Capture vectors, check the DoIO vector and that's it for memory. Just wipe the file off the disk and warn the User about the startup-sequence.

HAWNES:

A simple file type virus. It infects the OpenLibrary() vector waiting for an opening of intuition.library. It then patches OpenWindow() to it's own routine. When a window opens it checks the startup-sequence and if not already present, copies itself to the disk using DOS. It patched the VertB int and will display something after a while. It will Wipe out a disk after so many copies as well. Simple to remove and alter the first line in your startup-sequence which will hold in hex $C0A0E0A0C0.

RETURN OF THE LAMER:

Another file type virus which replaces the Disk-Validator. It uses a RomTag to stay in memory, infects vectors, VertBlank Int, trackdisk.device BeginIO(), and another vector in the trackdisk. When the RomTag is called it infects the OpenWindow() vector. Just delete the Disk-Validator and replace it from a good disk. In memory, just restore the vectors and clear the RomTag out.

TRAVELLING JACK:

A Link type virus this one installs itself into the internals of AmigaDOS taking over the BCPL inner workings. To check in memory we have to wind our way thru many vectors and then reinstall the original from the virus. To remove from file we just remove the first code hunk. Seems to copy itself to each file that has been read but not sure on this.

LIBERATOR:

This is a file virus that says it will remove all viruses but is in fact a virus itself. It copies itself to the s/startup-sequence with a line that says 'memcheck s'. You will also find a file called .FastDir on the disk. After a certain count it will

```
        delete the entire s/startup-sequence and display a message,
        and stop access to the floppy drives and DH0:.  It will also
        stop most virus checking programs by RemTask()ing them.
        Virus_Checker 5.30 and above is safe from the current version as
        the Task name has been changed.  Easy to remove, we just
        delete the file.

MENEM'S REVENGE:
        This is a Link virus. It starts a Task called a single space. This
        task sole job is to Patch the LoadSeg vector in DOS. It thus
        infects programs that are run. It is triggered thru the Amiga's
        time and will write it's message to files on dh0: and/or df0:
        The message it writes and then displays as an Alert is

        Menem's Revenge has arrived
        Argentina still alive

        All VC does is Remove the Task and reset the LoadSeg vector.
        It will be removed from files as they are scanned.
        It adds 3076 bytes to each file it infects.

        There are some problems with this virus. It does not know Amiga
        Files very well and will sometimes get it wrong. VC will remove the
        virus okay but the program may still not work due to this.
        Also when the virus is removed from memory the computer may lock-up
        or GURU after a while. This may be due to memory not being freed
        properly when the task is removed.

TRABBI:
        Harmless link virus. Will try to link itself into all files it can
        get at. Uses the drive it was started from. Creates a Task that
        will play a tone/music and put up a requester after a delay.

METAMORPHOSIS:
        This one is a combination Link/BootBlock virus. It will pick random
        files in the c: directory to infect when every the OldOpenLibrary()
        call is made. It also infects the DoIO() vector and this will write
        the bootblock part of the virus.
```

1.20 Virus_Checker Version Notes

```
*************************************************************************
                      Virus_Checker Version Notes
*************************************************************************

Version History shortened

6.00 Released 3 February 1992
     New Workbench 2.0 interface added and hotkey support put in.
     VC now supports the -l and -t commands again.  Don't know why they
     stopped working actually.
     Bug With Screen title.  When used with 2.0 Workbench and no shell,
     screen flashed allsorts of colors.  Have removed the Screen Title from
     VC.  Will do it a different way later on.
     Bug in checking of BootBlocks.  Was not getting error properly from BB
```

read. Thus if an error occured VC would use the Data from the last BB.
Hopefully this problem is now gone.

6.01 Released 24 February 1992

Added second row of drive gadgets to Interface Window. These will check all files automatically when disk is inserted.
Changed way I disabled the gadgets when VC is busy with scanning. Now uses a Request() to disable them.
Bug in VC Requester when it can't remove a link virus due to problems with the file. The Requester it put up could not be read as it was all squashed up.
Added Intuition EasyRequester when under WB2.0 and just a normal requester needed.
Small bug in checking files for link viruses. If file was corrupted in a certain way VC got into a loop it could not get out off. Have put in bounds checking to stop it reading outside it's buffer.
It seems the StayResident bit did not work. Have it fixed now.
Added Icon support. You can now configure VC via the icon if started from Workbench and CLI. (see THE WORKBENCH STARTUP:)
Finally found the trouble with the backdrop option. When used under WB2.0 the window will get hidden under the screens title bar.
Found the trouble with VC not letting computer change Font or screenmode and fixed it.
Added -i option to cmd line so that VC won't put up a requester saying it could not read the bootblock if there was an error.
Added CHECKFILE to arexx side for BBS use. This will take a file, check and remove any viruses found. It will then return a result to the calling program saying file was infected.

6.02 Bogus version around so I skipped this one

6.03 Released 30 March 1992

Forgot to free up the VisualInfo when VC opened it's window.
Didn't seem to make much difference. Maybe lost a few bytes of memory
Gave up on using Exit() to end Process after warnings in new AmigaDOS manual.
Small bug in WarmCapture checking code. If you canceled the requester VC just kept asking you if you wanted to clear it.
New version of Macro68 with it's new Optimization saved about 500 bytes of Memory. Also Sped up code quite a bit.
Added a new window that comes up during file scan. If any Viruses are found then the window will wait for you to click on a gadget to continue
Added a new feature. Under WB2 you can VC to watch the s:startup-sequence for any changes. VC will warn you when this happens.
Redefined default hotkey as Left-Amiga Shift Del as the previous one clashed with CygnusEd Write function.
FileName buffer was a bit too small. Some people must have really long directory/filenames.

6.04 Released 8 April 1992

Added Menem's Revenge Virus. Sent to me from the US it was spreading fast so I thought I had better get this out.
See virus notes on what it does and some bugs.

6.05 Released 12 May 1992

As WB2.0 will not allow Virus_Checker to read a file that is read protected I have put in some error checking. When doing a directory scan VC will warn you if the file is Read-Protected.
Small mistake in the WATCHSS from the workbench Start. Would not work due to a spelling mistake on my part

```
     Added CheckBootBlock\dfx: command to ARexx interface to check
     Bootblocks from scripts.
     Received Hacked version of Saddam Virus. Someone changed IRAK to LAME
     in code. Small change to VC to change these blocks back to normal
     Added Trabbi virus. Harmless link virus.
     Small bug in Disk Scan code when scanning single file. Called
     AllocMem() with a zero size.
     Added another version of SCA virus. This one just puts up a screen, no
     copying to other disks
     Added Bootblock viruses Triplex, Virus_V1_(Wieder_da), another version
     of Northstar, Sachsen, GeneStealer, Exterminator, and MG.
     Added Metamorphosis virus, this one is a link/BB virus
6.06 Released 4 June 1992
     Added Ignore BB Error configuration to Window.
     Removed other version of SCA virus. People did not mind it and it did
     not copy itself anyway.
     Heaps of bugs in virus removal in memory. Some viruses found as
     others. Could not test them due to not running on my machine. Did
     remove the viruses though.
     Finally got rid of that last ugly Requester under WB2.0
     Fixed a bug in the disk scan routine. If you entered a empty string in
     the string gadget VC should have checked the directory from which it
     had it's root. But it scan all sorts of silly directory names and
     would usually crash when done. Now fixed.
     Bug in rexx code for CheckBootBlock. You needed to have AutoCheck
     turned on otherwise it would not check the bootblock. Fixed.
6.15 Released 14 July 1992
     Version 6.07-6.12 skipped due to a fake 6.12 version
     Some more small bugs in memory detection taken out.
     VC will set the protection of File viruses so it can delete them
     Memory allocation 1 byte too small when adding virus names to scan
     window.
     Small bug with CheckBootBlock Arexx command. Would return Okay on some
     viruses
     Added Challenger, Disktroyer virus, Golden Rider, BlueBox,
6.16 Released 17 October 1992
     Added ReadArgs() for users with WB2.0 Read Above for command line
     Removed a couple of more bugs. Still 40 bytes of memory lost each time
     Virus_Checker is run.
     Added Infiltrator (Klein) link virus.
6.17 Released 19 October 1992
     Massive bug inflicted by stupid programming on command line.
6.18 Released 21 October 1992
     More bugs introduced due to fixing the previous bug. I hope this is it
6.19 Versions 6.15 and up had checking in them for hacking of code. In 6.16
     and above this is being found in most cases in the USA.  As I cannot
     afford the bill for all the E-mail the code has been dropped.  It will
     still check the version numbers.
6.20 Released 11 December 1992
     Added an option to Un-Protect read-protected files during file scan.
     Added two more versions of the Liberator File virus
     Added Cobra boot block virus and Sonja boot block virus
     Fixed bug with CloseWindow from Menu. I actually fixed this in the
     6.16 version I had before I lost the code.
6.21 Released 12 February 1993
     Fixed IGNOREBB not comming up checked when given on command line.
     Added the Unicorn, Adam Brierly BB viruses and added DStructure and
```

```
     Starlight file viruses
6.22 Released 14 Febuary 1993
     Added Timer file virus and fixed a serious loop bug.  Accidently put
     the wrong label in in one of the memory checks.  As a result and
     endless loop at priority 19 which made it seem the machine had locked
     up. Sorry guys.
6.23 Released 20 March 1993
     Added code to check crunched files. Uses Decrunch.library to do this
     and it must be in libs:
     Added Amiga Knight File virus, and Fake SnoopDos1.6 (a bbs backdoor)
6.24 Released 5 April 1993
     Altered file reading code so that file is read only once unless size
     changes between checks of Link/File viruses.
     Altered Interface to support new options. Redid most of support
     routines for GadTools gadgets. Some done illegally under WB3.0.
     Found and corrected a long standing memory loss bug.
     Check gadget now works properly under WB3.0
6.25 Released 19 April 1993
     Added QRLD Link virus and corrected a bug in link virus scanning code
     Altered way Virus_Checker handles input when it is already running.
     You can now use Virus_Checker quit (WB20) -q (1.3) to stop a running
     VC. Also it now longers sends messages to itself but uses another
     method to communicate to itself.
     Changed detection of Menems revenge in memory, was picking up replex
     as a virus.
     No longer asks if you wish to kill it but just pops the window open.
6.26 Released 17 May 1993
     Found bug in File virus delete routine. Would not always delete the
     file but did find virus and warn you.
     Added Kill option to delete ALL viruses in files (Link viruses as
     well)
     Added 6 versions of CompuPhagozyte file virus and SS bootblock virus
     Bug found in loading brainfile. Thanks to Eddy Carroll and SnoopDos
6.27 Test version only
6.28 Released 20 June 1993
     Corrected small bug where 6.25 and up would crash on some machines
     Mostly A600's affected
     Added new feature for checking archives. It is a one shot mode run
     only from CLI/SHELL and WB2.0x and up. See COMMAND LINE OPTIONS
     Added DM-Trash virus to checker
     Added InstallVC script. This uses CBM Installer program
     Added new special mode to WB2.0x command line. BBFILECHECK will force
     the file check to see if the file is a BootBlock virus. WARNING. This
     is a dangerous option and is only used for testing a BootBlock dump
     for viruses. Don't use it as it may detect good files as viruses.
6.29 Released 27 July 1993
     One more change for detection of menems revenge virus and replex. Some
     still saying it is being picked up but I cannot see how.
     Added option to disable CoolCapture not zero requester.
     Added support for the SHI's Bootblock.library. By using this library
     and its brainfile you have the ability to add new Bootblock viruses
     as SHI release new brainfiles. The BootBlock.brainfile goes in the
     L: directory and Bootblock.library in LIBS:. See BBLIB in docs
     Added New version of SKick into Capture check.
     Found small bug in WB support where VC would not recognise the YES
     command in the icon.
6.30 Released 17 August 1993
```

```
     Changed way in which Virus_Checker reads low memory. Will now only
     cause 2 Enforcer hits.
     Corrected bug that prevented BBLIB shell option not to work.
     Updated to latest BootBlock.library and Brainfile which adds more viruses
     Added more Bootblock viruses to internal code as well
     Added FU?K virus to code

*****************************************************************************
```