

This DOS Virus Toolkit consists of the following programs designed for disinfecting viruses on DOS files (**not compatible** with 64-bit Windows versions) as a last resort:

- **RVK: ROSE SWE Virus Killer** - A generic virus remover specifically designed for COM files.
- **DECOM: A generic decryptor for COM files.** For finding text strings or analysis
- **RPCATCH: ROSE SWE Poly Virus Catcher** - A heuristic virus scanner for detecting polymorphic viruses.



Written and (C)opyright 1992-2026 by ROSE SWE,
Dipl-Ing. Ralph Roth – See ROSEBBS.TXT for full address

1 Index

1 Index.....	1
2 Overview.....	3
3 RVK – Synopsis.....	3
4 About RVK.....	4
5 About The Cleaning Process.....	4
6 Multiple Infections and Anti-Debugging Tricks.....	5
7 Decom – Synopsis.....	5
8 About Decom.....	5
9 RPCatch.....	7
9.1 RPCatch – Usage.....	7
9.2 Parameters.....	7
9.3 Detection ratio.....	8
10 A Little Warning.....	8
11 Legal Terms and Disclaimer.....	8
12 RVK+DECOM – History.....	9
12.1 Version 0.01-0.05.....	9
12.2 Version 0.10.....	9
12.3 Version 0.11.....	9
12.4 Version 0.13.....	9
12.5 Version 0.20 (March 95).....	9
12.6 Version 0.21 (April 95).....	9
12.7 Version 0.23 (December 95).....	9
12.8 Version 0.24 (February 96).....	9
12.9 Version 1.20 (March 96).....	10
12.10 Version 1.21 (March 96).....	10
12.11 Version 1.22 (April 96).....	10
12.12 Version 1.23 (July 96).....	10

12.13 Version 1.24 (December 96).....	10
12.14 Version 1.25 (10 August 1997) and version 1.26 (15 February 1998).....	10
12.15 Version 1.29 (April 2002).....	10
12.16 Version 1.30 (May 2003, May 2011, 2013).....	10

2 Overview

DECOM/RVK/REG is a versatile and powerful tool designed for comprehensive DOS virus handling and protection of DOS COM files. The key capabilities of DECOM/RVK/REG include:

- **Virus Removal:** DECOM/RVK/REG effectively eliminates a variety of viruses, including but not limited to CryptCom, ICE, Scramble, RCrypt, HD-Killer, Crypt, and Shrink. This makes it an essential tool for maintaining the cleanliness of COM files and preventing malware-related issues.
- **Decryptor:** The tool decrypts encrypted COM files, enabling detailed analysis and examination. This feature is crucial for understanding the behaviour and structure of encrypted programs.
- **Polymorphic Encryption Handling:** DECOM/RVK/REG is particularly useful for studying and unpacking polymorphically encrypted COM programs or viruses. This capability is vital for researchers and cybersecurity professionals dealing with complex, mutating threats.
- **Antidebugger Handling:** It includes sophisticated routines to manage antidebugger techniques, particularly for systems running on 386+ processors. This feature helps in bypassing protections that prevent debugging, facilitating deeper analysis.
- **Code Emulator:** DECOM/RVK/REG features a built-in code emulator, allowing for the execution of code in a controlled environment. This emulation capability is essential for safely analysing potentially malicious code without risking the integrity of the host system.
- **80586/MMX Support:** The tool supports the 80586/MMX processor, ensuring compatibility with a wide range of systems and enhancing its utility in various computing environments.
- **Encryption Guesser:** DECOM/RVK/REG includes an encryption guesser, which aids in identifying the encryption algorithms used in encrypted COM files. This feature is particularly useful for cybersecurity experts tasked with decrypting and understanding obfuscated code.

Overall, DECOM/RVK/REG is an indispensable tool for professionals involved in virus analysis, decryption, and protection of COM files. Its comprehensive feature set ensures that users can effectively address and mitigate threats posed by a wide array of malicious programs.

3 RVK – Synopsis

The RVK utility is designed to meticulously handle both polymorphic (e.g., MtE, NED, DSME, DSCE, ViCE, TPE, SPE, G2, PS_MPC, etc.) encrypted DOS COM files and ordinary (unencrypted) viruses. This robust tool performs the following critical functions:

- **Decryption:** RVK systematically steps through the encrypted DOS COM file, decrypting its contents to reveal the underlying code.
- **Virus Removal:** It identifies and eradicates viruses from the infected file. This involves:
 - **Restoration:** Restoring the host program to its original state before infection.
 - **Virus Disabling:** Disabling the virus to prevent it from executing.
 - **Virus Removal:** Removing parts of the virus code from the infected file to ensure complete disinfection.

- **Prevention of Execution:** After completing the decryption and cleaning process, RVK terminates its operation before executing any part of the virus, ensuring that the virus does not get a chance to run.

In summary, RVK is an effective utility for decrypting, cleaning, and securing DOS COM files from polymorphic and ordinary viruses, ensuring the integrity and functionality of the host programs without risking the execution of malicious code.

4 About RVK

This program is useful if you have an infected file and you want to remove the virus. Just clean it using RVK, then check the resulting file. RVK isolates viral code in an infected program and disables it. From then on it will be safe to use the program again, as the risk of other files being infected or damaged by it will have been securely disabled.

5 About The Cleaning Process

RVK works completely different compared to the 'conventional' cleaners. First of all, it does not recognise any particular virus. However RVK is aware of many tricks used by common viruses. Its disinfection scheme is therefore completely different from known cleaners and it works with almost any (COM) virus. This technique is called heuristic cleaning mode! In that cleaning mode RVK does not need any information about viruses either, but it has the added advantage that it does not even care about the original, uninfected state of a program. This cleaning mode is very effective if your program is infected with an unknown, a polymorphic or with a virus using 80386+ instructions!

Note that this does not imply that the cleaned file is 100% equal to the original one. When RVK uses heuristic cleaning to disinfect the program, the file will never be exactly the same as in its original state. This is not an indication of failure of RVK, nor does it mean the file is still infected in some way. First of all, it is normal that the heuristic cleaned file is still larger than the original one. This is normal because RVK tries to be on the safe side and it will avoid removing too much from the host program. The bytes left at the end of the file are 'dead' code, the instructions will never be executed again, since the 'jump' at the beginning of the program has been removed. The functionality of the cleaned file will nevertheless be the same! For this reason a virus scanner MAY find still the virus in cleaned files – or will now report a new variant of this virus (F-Prot)!

In the heuristic mode, RVK loads the infected file and starts emulating, simulating and tracing the program code to find out which part of the file belongs to the original program and which to the virus. The result is successful if the functionality of the original program is restored, and the functionality of the virus has been reduced to zero. When used, RVK will attempt to follow the execution of the program until the end of the decryptor or if the original entry point is restored by the virus! It will not execute dangerous interrupt calls, and will terminate if one is encountered. Some interrupt calls will be simulated, some emulated, a few will be executed (e.g. "get DOS version" or virus installation check) and some will be removed! It also terminates if DS and ES change, or if a far call is encountered. **THIS DOES NOT ABSOLUTELY GUARANTEE SAFETY WHEN RUN!** The viruses I have tested RVK on are over 600 COM infectors! One possible time when RVK may go to pass the cleaning process is when the virus does not actually restore the host program – instead trying to go resident or to infect other victims. Please send me any virus that can not be killed with RVK! If possible I will improve RVK to clean this virus too.

6 Multiple Infections and Anti-Debugging Tricks

The process of removing viruses from infected files can indeed be quite complex, especially when dealing with multiple viruses or multiple instances of the same virus. This complexity is compounded by the fact that some viruses continuously re-infect files, leading to an increase in the size of the infected files over time. An example of such a virus is the Jerusalem virus, which is notorious for its ability to keep infecting files repeatedly. When using a tool like RVK it is crucial to understand its capabilities and limitations. RVK may only remove one instance of a virus at a time, necessitating repeated scans and cleaning until RVK confirms that no further infections are present. This iterative process ensures that all instances of the virus are removed, even if it requires multiple cleaning cycles.

Certain types of files, particularly COM files protected with anti-debugger techniques like RoseCrypt or HackStop, present additional challenges. These protections are designed to thwart debugging and analysis, making it difficult for RVK to clean them. In contrast, encryption added by tools like SCRAMBLE, CRYPTCOM, or R-Crypt can be safely removed by RVK, facilitating the cleaning process for files protected by these methods.

RVK's ability to bypass many of the anti-debugger tricks used by existing viruses, packers, and scramblers is a significant advantage. This capability allows RVK to effectively handle a wide range of obfuscation techniques that viruses use to evade detection and removal. However, users should be aware that some advanced protection mechanisms might still pose challenges.

To summarise, the key steps in effectively cleaning infected files using RVK include:

- Multiple Scans: Repeatedly running RVK until it reports no further infections, ensuring all virus instances are removed.
- Understanding File Protections: Recognizing that certain anti-debugger protections may prevent RVK from cleaning some files.
- Leveraging RVK's Capabilities: Utilizing RVK's ability to bypass many anti-debugger tricks to handle various obfuscation techniques.
- By following these steps, users can maximize the effectiveness of RVK in cleaning infected files and minimizing the risk of re-infection.

7 Decom – Synopsis

This is a simple utility that will step through a polymorph (MtE, TPE, SPE, G2, PS_MPC...) decryptor and decrypt the virus it is attached to, then terminate before executing the virus.

8 About Decom

It is useful if you have a (polymorph) encrypted virus and you want to find out what virus has infected it – just decrypt it using DECOM, then check the resulting file, looking after the decryptor. This is a prototype version, and is **NOT IN ANY WAY GUARANTEED!**

I had only released this program because to this date nothing else seems to be able to do this (apart from TBCLEAN, which removes the virus!). This will allow anyone who needs to be able to disinfect or to evaluate (polymorph) encrypted viruses. Afterwards you can modify the code to -instead of saving the result to disk- search it for the storage bytes, ori-

ginal SS:SP and CS:IP, or whatever is needed for the disinfection routine. A generic disinfectant (RVK) based on DECOM is also available...

When used, DECOM will attempt to follow the execution of the program until the end of the decryptor. It will not execute dangerous INT calls, and will terminate them if one is encountered. It also terminates if DS and ES change, or if a far call or something else is encountered that will cause the loss of control over the program's execution. **THIS DOES NOT ABSOLUTELY GUARANTEE SAFETY WHEN RUN!** While I have not encountered a polymorph encrypted file that it did not safely decrypt, it is quite possible to program such. The 'true' polymorph viruses I have tested DECOM on are:

- Alive:SPE
- Argyle
- Bosnia:TPE.1_2
- Byway (Dir-2.TheHndV)
- CoffeShop:MtE.0_90
- CoffeShop:TPE.1_0
- CoffeShop:TPE.1_3
- Connie:DSME
- Crazy_Chemist:SPE
- Dedicated.A:MtE.0_90
- Dedicated.B:MtE.0_90
- Dedicated.CryptLab:MtE.0_90
- Demo:DSCE
- Demo:DSME
- Demo:GCE
- Demo:PME
- Demo:SPE
- Demo:TPE.1_4
- EbbelWoi.QUX
- Encroacher.A:MtE.0_90
- Encroacher.B:MtE.0_90
- Fear:MtE.0_90
- Flip.2153.A
- Flip.2153.B
- Flip.2153.D
- Flip.2153.E
- Flip.2343
- Flip.2365
- GOL-Wanted
- Gotcha.Pogue:MtE.0_90
- Groove:MtE.0_90
- Insufficient.A:MtE.0_90
- Insufficient.B:MtE.0_90
- Insufficient.C:MtE.0_90
- King:SPE
- Lame:DAME.0_91
- Lame:HPE.0_90
- Lame:HPE.0_91
- Little:TPE.1_3
- Ludwig.A:MtE.0_90
- Ludwig.B:MtE.0_90
- Ludwig.C:MtE.0_90
- Natas.4730
- Natas.4738
- Natas.4744
- Natas.4746
- Natas.4748
- Natas.4988
- N8fall (the 4xxx versions, as well „Won't last“, 57xx versions) - com files only...
- One_Half.3744 (fails sometimes)
- One_Half.3755 (fails sometimes)
- Ontario.1024
- PC_Weevil:MtE.0_90
- Phoenix.1226
- Phoenix.2000
- Phoenix.Evil
- Phoenix.Phoenix.A
- Phoenix.Phoenix.B
- Phoenix.Proud
- SMEG:Pathogen (too complex for DECOM!)
- SMEG:Trivial (too complex for DECOM!)
- Teacher:DSME
- Tester:NED.1_00
- Testfiles:TPE.1_0
- Testfiles:TPE.1_4
- Tremor (COM-Variant)
- Trigger:DAME.0_90
- Uruguay Family
- V2P6
- V2PX.1260
- WordSwap.1503

As well as a collection of my own MtE & TPE test files (15000!) and over 400 different encrypted viruses (Cascade, G2, PS-MPC, ANNI-VCS, IVP, VCL, etc.). One possibility when DECOM is not able to decrypt the code is:

- the decryptor does not actually encrypt the code
- the code is not encrypted in any way
- anti-emulator code is found
- the decryptor uses anti-debugging tricks, which DECOM is not yet aware of
- if there are „do nothing“ loops like sometimes found in the TPE 1.3/1.4 viruses. In this case use RVK!

This generally results in DECOM printing that it can not safely decrypt it. If you got the hands on such a file please send me it in order to improve DECOM.

9 RPCatch

I am the author of a German virus scanner called VirScan Plus (and other virus scanner like mpscan or RMS), which is able to detect more than 60.000 viruses. The most time I spend to add detection of polymorph viruses to VirScan Plus. For this reason I have written RPCatch, a generic heuristic scanner for encrypted viruses. Later, when RPCatch is stable, the routines will be incorporated into VirScan Plus. RPCatch has a build-in 80386 disassembler as well as a code emulator and a heuristic detection engine to catch all those polymorph encrypted viruses.

```
ROSE SWE's 'super' COM-Crypt/286 - Version 1.20 - FREeware - Released [03/29/13]
(c) 1995-2013 by ROSE SWE, Dipl.-Ing. Ralph Roth - See ROSEBBS.TXT 4 address
reg.com: reading, mutating (MTE1), garbling, writing, mutating (MTE2), inoculat
ion, done! ①
Loader=68, Mutated Loader=99, added protector=219, Total=287
Plain RSSC protector size is 151 bytes ②
c:\src\asm\decom>rpcatch.com ..

-----|-----
:  RPCatch | ROSE polymorph virus catcher (generic scanner) | Version 0.03  :
-----|-----
(C) 1993-2014 by ROSE SWE, Dipl.-Ing. Ralph Roth - http://rose.rult.at
Released as BANNERWARE/FREWARE. ALL RIGHTS RESERVED (ALLE RECHTE VORBEHALTEN)!
>>> Still alpha version! Contains minor bugs, like directory display! <<<

C:\...\SRC\ASM\DECOM\DECOM.COM
C:\...\SRC\ASM\DECOM\REG.COM polymorph structure! ③
C:\...\SRC\ASM\DECOM\RPCATCH.COM
C:\...\SRC\ASM\DECOM\RVK.COM

c:\src\asm\decom>
```

RPCatch detected an example file encrypted with RSSC (protector using two mutation engines).

9.1 RPCatch – Usage

Invoking RPCatch with no parameters will result in a recursively scan of the current directory. You can invoke RPCatch additionally with a drive statement with will result in a recursive scan from the root directory of the specific drive.

9.2 Parameters

/? -?	a short help
drive:	drive to be scanned

E.g.: rpcatch c:

9.3 Detection ratio

RPCatch detects about 100% of all Tremor, TPE, MtE and DSME encrypted viruses. Furthermore RPCatch detects almost all simple encrypted viruses such as VCL, PS_MPC, IVP, BW or G2. In generally spoken, RPCatch detects about 90% of all encrypted viruses I have in my collection (and that's a big amount). To be honest, RPCatch will also detect all encrypted programs with are protected by programs like CryptCom, Scramble or Protect.

10 A Little Warning

This package is a prototype version, and is **NOT IN ANY WAY guaranteed**! I am only releasing this program because to this date nothing else seems to be able to do this (apart from TBCLEAN which is disbanded meanwhile). This will allow anyone to be able to disinfect COM files. As an advantage RVK is not limited to 8086 code, it will even clean viruses which will use 80586+ instructions (remember: you CAN NOT CLEAN 386 code on a 286 machine)! Send me ANY virus that could not be cleaned by using RVK!

11 Legal Terms and Disclaimer

RVK+DECOM basically has no legal guarantee and warranty because I do not want to get sued over it, and should be used "as is". Here is the official disclaimer:

RVK+DECOM ("program") **will ALTER and DESTROY executable files** and may have or cause **compatibility problems** with them (that is why YOU should keep a backup file, in case of incompatibility with a particular file) in certain circumstances. Under no circumstances may Ralph Roth ("author") be held liable or accountable for any damage to system files, executable files, data files, or any other system or data damage due to use or misuse of his program. The author also may not be held accountable for loss of profits or for any other damages incurred by the use or misuse of his program. The author has forewarned any users that damage to files may occur with use or misuse of his program, and in executing the program, the user fully understands these risks and this disclaimer.

Greetings (and virus free time)
Ralph Roth

You can obtain the newest DECOM & RVK version from our home page – see ROSE_BBS.TXT

12 RVK+DECOM – History

12.1 Version 0.01-0.05

Now RVK prompts you only for a file name _IF_ the virus has been safely decrypted or disabled! This means although, that you can now overwrite the old file at your own risk. RVK no emulates a lot of MS-DOS calls to handle many more viruses! "Anti Debugger Code Handling" improved!

12.2 Version 0.10

RVK can now be invoked via command line else you will be prompted for a source file! RVK now truncates (most of) the virus-body, therefore check the resulting file carefully!

12.3 Version 0.11

Added more code checking in order to clean the Annihilator Stealth viruses. RVK displays now information about the cleaned file. Some (dangerous) instructions are now additionally overwritten with NOP's, therefore check your cleaned files carefully!

12.4 Version 0.13

Added more anti-debugging tricks checking. Tested with over 50 new viruses.

12.5 Version 0.20 (March 95)

Added a software emulator that is able to emulate INT calls and most anti debugger tricks without loosing control over the program! RVK can now handle almost all files, except some special anti debugging code. Furthermore the handling of infected files is now safer, more reliable and more successful than ever before!

12.6 Version 0.21 (April 95)

My FAX number has changed! Little code enhancements to clean more viruses! The program is now able to by-pass some IN/OUT commands. The package now includes an alpha version of the heuristic scanner "RPCATCH".

12.7 Version 0.23 (December 95)

Fixed some orthographical errors in RVK.COM.

12.8 Version 0.24 (February 96)

The code emulator can now handle the POP SS/POPF anti debugger trick. Credits goes to L. Vrtik & J. Valky for pointing out this trick, as well as supplying me sample code.

12.9 Version 1.20 (March 96)

Added the handling of 386++ commands. For this reason you will need at least a 386 SX to run the program! Changed the version number to 1.20 (now the same as DECOM). The code emulator can now handle another anti debugger trick. Credits goes to L. Vrtik & J. Valky for pointing out this trick, as well as supplying me sample code. Excluded the RPCatch program from the package.

12.10 Version 1.21 (March 96)

The code emulator can now handle another anti debugger trick. Credits goes to L. Vrtik & J. Valky for pointing out this trick, as well as supplying me sample code. Added the "TB-Clean Bug" from VLAD #6 to the emulator, as well as another anti debugger trick found in the GOL-Wanted virus, which hinder RVK to clean infected files. Credits goes for this goes to Martin Roesler.

12.11 Version 1.22 (April 96)

The code emulator can now handle GS: and FS: segment override anti debugging tricks. Credits goes to L. Vrtik & J. Valky for pointing out this trick, as well as supplying me sample code. Added the handling of protected mode debugging tricks, using the CR and DR registers.

12.12 Version 1.23 (July 96)

The code emulator can now handle the PUSHFD/POPFD anti debugging trick and other 32 bit anti debugger tricks. Credits goes to Rand0m^X-Adi for pointing out this trick.

12.13 Version 1.24 (December 96)

Minor small bug fixes. Fixed some typos in the DOC. Added an interrupt 3 emulator. Added code to handle anti-emulator code found in the Grief.3584 and ANNI-VCS viruses. Now the program displays the last IP Counter, the AX value and the opcode of the latest instruction if the emulating process failed. This is useful to find out why and where the emulations process has been interfered.

12.14 Version 1.25 (10 August 1997) and version 1.26 (15 February 1998)

Minor code and documentation changes. Version 1.25 was released on the VIRUS.GER CD-ROM (published by VHM). The Cicatrix cumulative update January 1998 contains this version along with tons of viruses. To avoid speculations if this is version was hacked or infected this new version is released instead!

12.15 Version 1.29 (April 2002)

Merged the documents into one big PDF file. Some small fixes on the source code.

12.16 Version 1.30 (May 2003, May 2011, 2013)

Enhanced the documentation. Some small fixes on the source code.

Please excuse my English; it is not my native language! :-)

Feedback and bug reports are appreciated!