



**VirusBuster**  
for Samba Servers  
1.2

## TABLE OF CONTENTS

<b>INTRODUCTION .....</b>	<b>2</b>
<b>VIRUSBUSTER FOR SAMBA SERVERS .....</b>	<b>3</b>
<b>System requirements .....</b>	<b>3</b>
<b>Minimal required Linux distributions .....</b>	<b>3</b>
<b>Supported Samba versions .....</b>	<b>3</b>
<b>Installation, uninstallation .....</b>	<b>4</b>
<b>Operation.....</b>	<b>4</b>
<b>Configuration settings .....</b>	<b>5</b>
General settings.....	5
Notification .....	7
Available tokens for messages .....	7
Registration data.....	7
Including external configuration file .....	7
<b>VIRUSBUSTER SCAN DAEMON.....</b>	<b>8</b>
<b>Executable files .....</b>	<b>8</b>
<b>Database update.....</b>	<b>8</b>
<b>Configuration files.....</b>	<b>9</b>
<b>END USER AGREEMENT .....</b>	<b>16</b>
<b>CONTACT .....</b>	<b>17</b>

## INTRODUCTION

The package includes the *VirusBuster for Samba Servers* and the *VirusBuster ScanDaemon* daemon programs.

The name structure of the package:

```
vbsambashield-<vbsambashield-version>-<vbscand-version>-<opsystem>.tgz
```

```
e.g.: vbsambashield-1.2.0-1.2.0-linux-i386.tgz
```

The 'vbinst-pkg.pl' is the main install script file, run it to install both the vbsambashield and vbscand packages.

Available install script parameters:

```
--no-update
```

The installer doesn't install the vbscand.

The installer tries to install the vbscand package first. If the installer finds an installed instance of the vbscand in the system and its version is less than the current version, it removes the older one and installs the new one (if it is not disabled by the --no-update parameter). After vbscand has been installed successfully, the script also installs or updates the vbsambashield.

## VIRUSBUSTER FOR SAMBA SERVERS

The program ensures comprehensive virus protection for Samba file servers running on Linux or Solaris operating system. The on access protection keeps the computer virus free. It works totally transparent mode, scans the files in the background, users don't perceive its operation.

The program is divided into two modules:

- shield module, which performs the scanning functions
- vfsemu module, which is an interface for the different Samba version

### ***System requirements***

Supported operating systems:

- Linux i386/amd64
- Solaris 9 (sparc), 10 (i386)

Requirements for all the supported platforms:

- 256 MB free memory
- 100 MB free hard disk space
- wget (for update)
- perl5 (for update)
- VirusBuster ScanDaemon

Requirements by platforms:

- Intel Pentium (or compatible) processor at 300 MHz (Linux, Solaris 10)
- Ultra Sparc IIe processor at 500 MHz (Solaris 9)
- Minimal required for Linux: GLIBC 2.2.5, kernel 2.2.1

### ***Minimal required Linux distributions***

SuSE 8.0  
RedHat 7.3  
Debian 3.0 (woody)  
Mandrake 9.0  
Slackware 8.1

### ***Supported Samba versions***

Samba 2.2.1 - 3.4.0

Remark:

Those Samba versions that are newer than the mentioned above may be supported if their VFS module interface is compatible with the earlier versions. The 2.2.1-2.2.3 versions are only supported if the Samba daemon's binary was built with exported symbols. 3.0 alpha XY versions are not supported.

## ***Installation, uninstallation***

The program is available in a tgz package:

Unpacking:

```
tar -xzvf vbsambashield-<version>-<platform>.tgz
```

```
Installation: vbsambashield-install.pl
```

```
Uninstallation: vbsambashield-uninstall.pl
```

## ***Operation***

To activate the virus protection, the user has to modify the Samba's configuration file.

Setting of the various versions is differ:

**In case of 2.x.y Samba version:**

```
vfs object = <vfs module's name with path>  
(specify absolute path)
```

**In case of 3.x.y Samba version:**

```
vfs objects = <vfs module's name with path>  
or  
vfs objects = <vfs module's name>
```

If this option already has parameter in the configuration file then the required line must be inserted to the previous ones.

The options above can be specified generally for all the shares (the option and its parameter must be placed in the [global] section) or for certain shares (the option and its value must be placed into the required share(s)).

Examples:

```
vfs object = /usr/lib/samba/vfs/vbsambavfsemu.so  
or  
vfs objects = vbsambavfsemu
```

The program scans for viruses only on that connection that have been established after the configuration file's modification, the previous ones will not be concerned.

## Configuration settings

The configuration file is located on the `/etc/vbsambashield/general.conf` path.

Settings in the `[general]` section

### General settings

#### **registration-username**

Enter user name.

#### **registration-key**

Enter registration key.

#### **scanaddress**

Scan daemon's network address.

There are two supported socket types: unix socket and internet socket.

unix socket syntax: `unix:<path>`

internet socket syntax: `inet:<hostname or ip address>:<port>`

Example: `scanaddress= unix:/var/run/vbscand`

#### **engine\_path**

#### **vdb\_file**

These options need to be set when we use direct connect to the virus scan engine (without using `scandaemon`) - this is the default setting. The `engine_path` option has to point to the virus scan engine file, the `vdb_file` option to the virus database file.

Example: `engine_path=/usr/lib/libvengine.so`

`vdb_file=/var/lib/vbuster/vdb.xml`

#### **killable\_action**

If the virus is killable one of the following actions can be performed.

Available values:

`delete` - deleting file

`kill` - killing virus

`rename` - renaming file

`skip` - ignore incident

Default: `kill`

#### **non\_killable\_action**

If the virus is non-killable one of the following actions can be performed.

Available values:

`delete` - deleting file

`rename` - renaming file

`skip` - ignore incident

Default: `skip`

#### **suspicious\_action**

In case of suspicious file is found, one of the following actions can be performed.

Available values:

`delete` - deleting file

`rename` - renaming file

`skip` - ignore incident

Default: `skip`

#### **scan\_method**

Specifying scanning method.

Available values: `strict/fast/full`

Default: strict

## **heuristic\_level**

Specifying the level of the heuristics scanning.

Available values:

normal - normal level

off - heuristics off

high - high level

Default: normal

## **access\_on\_error**

Allow or deny access to the file if error(s) occurred during the scan.

Values: allow/deny

## **log\_level**

Log level setting.

Available values:

EMERG or 0

ALERT or 1

CRIT or 2

ERR or 3

WARNING or 4

NOTICE or 5

INFO or 6

DEBUG or 7

DEBUG0 or 7

DEBUG1 or 8

DEBUG2 or 9

DEBUG3 or 10

DEBUG4 or 11

DEBUG5 or 12

DEBUG6 or 13

DEBUG7 or 14

DEBUG8 or 15

Default: INFO

## **file\_log**

Using log-file (yes/no).

Default: yes

## **log\_path**

Location of the log file. This file stores the log events created during the program's operation.

Default: /var/log/vbsambashield/general.log

## **virus\_log\_path**

Location of the virus-log file. This file stores the log events created during virus scanning.

Default: /var/log/vbsambashield/VirusScan.log

## **samba\_log**

Using Samba log-system (yes/no).

Default: yes

## **syslog**

Using syslog (yes/no).

Default: no

## Notification

**message\_sender = Samba Shield**  
Message sender's name.

**message\_virus\_killed = "Message"**  
Sent message in case of virus found and killed. It must be specified between quotes, you can use tokens in the text.

**message\_access\_denied = "Message"**  
Sent message in case of virus found but any error occurred. It must be specified between quotes, you can use tokens in the text.

The following settings must be specified to display the messages in case:

- using nt based Windows: running Messenger service
- Windows 9x: installed network message sender application (e.g. Winpopup)

## Available tokens for messages

**%virus%**  
Last found virus name.

**%file%**  
The scanned file's name.

**%version%**  
SambaShield version number.

## Registration data

**registration-username**  
Enter user name.

**registration-key**  
Enter registration key.

## Including external configuration file

It is possible to include external configuration files into the main configuration file. These linked files will be processed as a part of the main configuration file.

The architecture of external configuration files must be the same as the original, use the @ sign to specify external link to a file in the main configuration file.

Example:

**@messages.conf**

The default configuration file includes external file, too, linking messages sent in case of virus incident. It is practical to change the language of warning messages easily and quickly.

## VIRUSBUSTER SCAN DAEMON

The VirusBuster ScanDaemon (hereinafter called scandaemon) provides an interface for the remote client program to utilize the full functionality of the virus scan engine through unix or internet socket. The package also includes a command line scanner client program (vbscan).

To establish the connection between the scandaemon and its client(s) you have to set a common communication address. Use the 'address' option in the configuration file of the scandaemon to set this address.

When the client is started, it tries to connect to the scan daemon and if it fails it displays an error message.

### **Executable files**

#### Parameters of the scandaemon

-----

##### **vbscand [options]**

Scan daemon binary file. Possible options:

-n, --nodaemon - no daemon mode (run in the console where started)  
-v, --version - display the version of vbscand and exit  
-b, --build - display the version and build of vbscand and exit  
-c FILE, --config=FILE - read configuration from FILE  
-p FILE, --pid\_file=FILE - save the pid to FILE  
-d FILE, --vdb\_file=FILE - virus database descriptor file  
-a ADDR, --address=ADDR - connect to ADDR  
-k SEC, --conn\_timeout=SEC - connection timeout in seconds  
-r SEC, --read\_timeout=SEC - read timeout in seconds  
-w SEC, --write\_timeout=SEC - write timeout in seconds

#### Parameters of Scan daemon init script (found in the /etc directory)

-----

##### **vbscand [options]**

start - start scan daemon  
stop - stop scan daemon  
restart - restart scan daemon  
cfgreload - reload configuration file  
vdbreload - reload virus database

#### Command line scanner client

-----

##### **vbscan [options]**

The available options are described below where the configuration file is detailed (vbscan.ini).

### **Database update**

You can update the virus database manually or automatically by the updater script found in the package.

The product uses incremental virus database update mechanism to keep the virus database up-to-date. The advantage of this method is that the program doesn't need to

download the whole virus database file every time (its size is several MBs) but usually only a small additional database package including the virus signatures processed and released recently. Using this mechanism, the download time is decreased to a minimal level so we can release additional virus database packages several times a day to improve the defense. Users can obtain protection against new malware without spending long time and generating considerable network load for the update. The protection is available almost immediately after the signatures of the newly discovered viruses have been processed in our virus lab.

## Automatic update

-----

We create a script to automate the update process, it is in the /usr/bin directory (vbs\_vdbupdate.sh).

Execute it, it is going to download the virus database, copies it into the correct directory and activates it. Updating will only be performed, if the database available on the server is newer than one on your computer. Otherwise the database will be left unchanged.

To execute the script, you should enter the vbs\_vdbupdate.sh command. It is possible to use parameters, too:

```
-f, --ftp          uses FTP update source
-h, --http,       uses HTTP update source
-v, --verbose,    verbose mode
--help,          displays help
```

### Example:

```
vbs_vdbupdate.sh -v --http
```

It downloads the database from the HTTP source and the progress bar will be displayed.

To run the script, you need wget program! By the help of cron, you can schedule the script executing to be performed by half an hours. Register into /etc/crontab:

```
0,30 * * * * root /usr/bin/vbs_vdbupdate.sh
```

## Manual update

-----

Our virus database-set consist of several files, you need to update all the files from our FTP server from the following folder and copy them to the virus database folder (/var/lib/vbuster):

```
update.virusbuster.hu/pub12/vbuster/vdb12/
```

You can activate the new database by the "vbscand vdbreload" command.

## Configuration files

### ----- Configuration file of Scan daemon (vbscand.conf) -----

The configuration file stores the settings in hierarchical structure. The storing mechanism based on the encapsulation concept which means that user has to specify the storing path (section) for each coherent setting group step by step.

The path (section) must be specified between square brackets in the configuration file:

```
[General]
```

Enter comments by using semicolon (;) before the comment text. The characters entered after semicolon will not be interpreted by the parser.

The configuration file ('vbscand.conf') can be found in the /etc/vbuster directory by default.

Options to be set in the [General] section:

**address=unix:/var/run/vbscand**

Address to listen on.

Two socket types are supported: unix socket and internet socket.

unix socket syntax: unix:<path>

internet socket syntax: inet:<hostname or ip address>:<port>

Default: unix:/var/run/vbscand

**vdb\_file=/var/lib/vbuster/vdb.xml**

Path to virus database descriptor file.

Default: /var/lib/vbuster/vdb.xml

**pid\_file=/var/run/vbscand.pid**

Path to pid file.

Default: /var/run/vbscand.pid

**conn\_timeout=1**

**read\_timeout=180**

**write\_timeout=1**

Socket timeout in seconds.

conn\_timeout: connection accept timeout

read\_timeout: socket reading timeout

write\_timeout: socket writing timeout

Default value is 1 to all options.

-----  
**Configuration file of the command line scanner (vbscan.ini)**  
-----

The configuration file is line-oriented for simple handling. Each line contains different settings, the option's name is conform to long option names.

You can use both the one character long- and the more character long options without dash ('-' or '--'). The options' values are similar to the command line options. In case of logical options the presence or the lack of the related option specifies if the function is enabled or disabled similar to the command line specification. Comments can be specified after '#' character in a new line or at the end of an opened line.

The configuration file (vbscan.ini) can be found in the /etc/vbscan directory by default.

**Connection**  
-----

**--attach**

Scan daemon's network address.

There are two supported socket types: unix socket and internet socket.

unix socket syntax: unix:<path>

internet socket syntax: inet:<hostname or ip address>:<port>  
Example: attach= unix:/var/run/vbscand

## **--engine**

### **--vdb**

These options need to be set if we would like to connect directly to the virus scan engine (without using scandamon). The engine option has to point to the virus scan engine file, the vdb option to the virus database file.

Example: engine=/usr/lib/libvbengine.so  
vdb=/var/lib/vbuster/vdb.xml

## **Information**

-----

### **-V --version**

Prints the version number of the program, scan engine and vdb.

### **-h --help**

Prints the general command line options, their default values and the application's version number.

### **--full-help**

Prints all the command line options, their default values and the application's version number.

## **Registration data**

-----

### **-k --registration-key**

Specifying the registration key based on your license. The program handles the hyphen separated form, too (XXXXX-XXXXX-XXXXX).

### **-u --registered-user**

Specifying the user name based on your license.

Using program without - valid - registration data you have to wait 30 seconds after starting scanner. You have to specify both the registration key and the user name for successfully registration.

## **Operational settings**

-----

### **-T --terse**

Enables compact log mode.

Compact mode:

```
/mnt/test/eicar.zip//eicar1.com: found: EICAR skipped.
```

Original mode:

```
/mnt/test/eicar.zip//eicar1.com
```

```
    virus found: EICAR_test_file (NOT killable) ... skipped.
```

### **-TT --terse --terse**

Only found information will be logged.

### **-q --quiet**

Enables the quiet working method. The program displays only the virus incidents on the screen or in the log file and a summarized statistics at the end of the scan.

Duplicate use:

### **-qq or --quiet --quiet**

This time the program writes out just the virus incidents to the stdout, summarized statistics also skipped.

Triple use:

## **-qqq or --quiet -quiet --quiet**

Combines the effect of --terse and -qq options.

IMPORTANT! 'quiet' option affects only the stdout, error messages could be returned by stderr.

## **-qqqq or --quiet --quiet -quiet --quiet**

If there is no need to notify the user (there was no malware found during the scan), entries will not be created into the log file.

## **--summary**

Disables displaying summarized statistics tables about the scan.

If -qq or --quiet --quiet options are also specified, it results reversed action: enables the summary display.

## **-o --old**

The program doesn't show warning message if virus database is older than two weeks.

## **-c --config=FILE**

Specifying the used configuration file with its path. If this option is not set, the program is looking for that file (named vbscan.ini by default) in the actual folder. If that one doesn't contain the .ini file the program's home directory will also be scanned for it. The configuration file is suitable for storing common settings needed for a general scanning. These settings can be redefined by command line options if necessary.

Note that the program will try to locate the files and directories that are specified by relative path in the configuration file starting from the actual directory (from which one the program was launched).

## **--debug=FILE**

If debug file is specified, the program will create it during the scan process to log detailed information about the program operations. It can help you to analyze the scan if necessary.

## **Scan area settings**

-----

## **-Z --skip-archive**

Archived files will not be scanned.

The archived files are scanned by default.

## **-M --skip-mail**

MIME of type files will not be scanned.

The MIME files scanning is included by default.

## **--symlink=ACTION**

Handling symbolic references (this option is working only on unix systems).

Available values (actions):

follow - uses the link name to identify the file

resolve - uses the file's own name to identify it

(in such a cases, it scans the referenced file as a regular file)

skip - ignores symbolic references

(in such a case it doesn't scan symlinks)

## **-R --skip-subdir[=PATH]**

The program scans each subdirectories recursively by default if a directory is specified as target. If you set this option, you can select directories or directory fragments to exclude from the scan while the other locations will be scanned recursively. If you use this option without parameter (the -R or --skip-subdir alone) then all the subdirectories of the specified target area will be ignored.

## **-f --file=FILE**

Text file containing paths and files (objects) to be scanned. This option's value locates the path of this text file. The objects will be read by lines from the file.

Special parameter: '-' hyphen ('--file=-'): this time the scanner reads the names of the files or directories to scan from STDIN (only in automatic mode).

## Scanned file types

-----

### --all-files

Switches off the pattern matching at all so all file types will be scanned.

### -p --pattern=PATTERN

The program scans only that files which match the specified PATTERN.

### --include=PATTERN

Adds PATTERN to the default configuration.

### --exclude=PATTERN

Excludes PATTERN from the default configuration. This option takes precedence over the above options.

### -m --match-in-archive

Pattern-matching inside the archives is enabled by default if you use the built in patterns of the scan engine for scanning (using '--include' and/or '--exclude'). In every other cases it is disabled (using '--pattern' or '--all-files'). This option changes the default value.

## Relations:

- '--all-files', '--pattern', '--include' could not be used at the same time
- If you don't specify either of the above options, the program will scan the files with the default extensions

## PATTERN syntax:

Several patterns can be specified in the pattern option separated by pipe (|). The pattern can contain ? and \* meta-characters (the ? (question mark) is considered as an optional character, the \* (star) is considered as an optional character chain). The program is also able to handle character-classes for more restriction. Character-classes must be specified between brackets (e.g. [abc]). The exclamation mark '!' means negation if it is placed straight after the initial bracket '['. The '-' sign placed between two characters means a character range. If you want the '-' or '!' signs to be a considerable character, you should place it straight before the ending bracket ']'. The program does not make a distinction between small and capital letters.

The '\*' and '?' meta-characters do not match directory separator characters in pathnames. The special '\*\*' sequence can be used to match any arbitrary characters including directory separators. For example:

'Program Files\\*\*.exe' - each .exe file will be matched in the Program Files directory and its subdirectories

### Important!

PATTERN is matched against file's basename (filename without path) if PATTERN itself does not contain directory separator characters or '\*\*' sequences, otherwise full path to the file shall be used. The separator character is '/' on Unix and GNU/Linux, and '\' on Windows that is the same as you can use to convert meta characters to literals. The application usually consider '\' as directory separator. It should be used duplicated if special characters follow it. These characters are: | \* ? [ ] .

## Scanning methods and actions in case of virus incidents

-----

**-e --heuristics = ( o | off | n | normal | h | high )**

Heuristics level setting. Default: normal level.

o / off - heuristics off

n / normal - normal level

h / high - high level

**-s --scanning = ( fa | fast | s | strict | fu | full )**

Scanning method setting. Default: regular level.

fa / fast - Only scans those parts of the file, which are most likely to contain a virus and does not detect viruses, which can only be detected by using a major amount of system resources (e.g. Excel FORMULA viruses).

s / strict - Optimized scanning method, which detects all viruses registered in the virus database and scans those parts of the file, which are most likely to contain a virus.

fu / full - Detects all viruses registered in the virus database and scans the whole file, even those parts, where viruses are not likely to be found.

**--thread=NUM**

Maximum number of program threads. This option's value is 1 by default. The multi-thread applications result in better performance, but this strongly depends on the system settings.

**--timeout=NUM**

Timeout limit of the scanning threads (seconds). Scanning will be cancelled if all the threads or just one of them exceed this limit - depending on the `--timeout-abort` option - and have no activity over the specified time-interval. You should increase this limit in case of large archives or strongly loaded system.

**--timeout-abort**

A `--timeout-abort` option affects the abort mechanism. If this option is set the program will be aborted immediately in case of first timeout.

This function is disabled by default that means the program runs until at least one thread ends within the specified limit.

The default `--thread` setting allows only one thread to be run so if it exceeds the timeout the program will be aborted.

**-a --action=ACTION**

Setting this option the program can be run in automatic mode so the specified action(s) will be performed without user interaction on virus incidents. If the action option is used repeatedly in the command line (separated by commas (,)), the actions will be considered and performed by their order. The first specified action has the highest priority and so on. If the first action can't be performed the following one will be tried. If the action (`-a` or `--action`) is not specified at all, the user is asked to choose an action in case of any incidents (interactive mode). Meaning of the available actions.

k - virus killing from the file (kill)

s - ignores the infected object (skip)

r - renaming the file (rename)

d - irreversible deleting (delete)

**--remove-macro**

Automatically deletes all the macros from the Microsoft Office documents without any confirmation.

**-G --greyware**

If you use this option, the program will detect the applications marked as greyware in the database and perform the specified action on them.

Greyware cannot be clearly categorized as malicious or not malicious application because it strongly depends on its use. Generally this kind of software is not harmful program in case it is installed by the user's consent and approval. But it can happen, that this program is installed in the background without the user's permission and in this case it can be used for malicious activity (for example an ftp server program or a remote access application).

So, in case of greyware, we cannot declare the application as malicious or not malicious based on the name or files of the program, it depends on the method of its installation.

## File- and directory references

-----

### **-d     --vdb=DIR**

Specifying the location of the XML descriptor file of the virus database. It is not compulsory to use this option but recommended. If the value of this option is not set, the program will be looking for the virus database in the 'vdb' folder of its home directory.

### **--log[=FILE]**

Screen output could be saved into a specified log file. If file name is not specified (FILE), the output will be appended to the end of a possibly available log file with the default name (vbscan.log).

## END USER AGREEMENT

*THIS SOFTWARE END USER LICENSE AGREEMENT ("EULA") IS A LEGAL AGREEMENT BETWEEN YOU AND VirusBuster Ltd. READ IT CAREFULLY BEFORE COMPLETING THE INSTALLATION PROCESS AND USING THE SOFTWARE. IT PROVIDES A LICENSE TO USE THE SOFTWARE AND CONTAINS WARRANTY INFORMATION AND LIABILITY DISCLAIMERS. BY INSTALLING AND USING THE SOFTWARE, YOU ARE CONFIRMING YOUR ACCEPTANCE OF THE SOFTWARE AND AGREEING TO BECOME BOUND BY THE TERMS OF THIS AGREEMENT. IF YOU DO NOT AGREE TO BE BOUND BY THESE TERMS THEN DO NOT INSTALL THE SOFTWARE.*

*IMPORTANT NOTICE TO USERS: THE SOFTWARE IS NOT FAULT-TOLERANT AND IS NOT DESIGNED OR INTENDED FOR USE IN ANY HAZARDOUS ENVIRONMENT REQUIRING FAIL-SAFE PERFORMANCE OR OPERATION. THIS SOFTWARE IS NOT FOR USE IN THE OPERATION OF AIRCRAFT NAVIGATION, NUCLEAR FACILITIES, OR COMMUNICATION SYSTEMS, WEAPONS SYSTEMS, DIRECT OR INDIRECT LIFE-SUPPORT SYSTEMS, AIR TRAFFIC CONTROL, OR ANY APPLICATION OR INSTALLATION WHERE FAILURE COULD RESULT IN DEATH, SEVERE PHYSICAL INJURY OR PROPERTY DAMAGE.*

### 1. Definitions

- (a) "Educational Version" means a version of the Software, so identified, for use by students and faculty of educational institutions only. "Home version" means a version of the Software, so identified, for use by individuals on a single computer at home only. Educational and Home Versions may not be used for, or distributed to any party for, any commercial purpose.*
- (b) Henceforward VirusBuster Ltd. means VirusBuster Ltd. and (where interpretable) its suppliers and licensors, if any.*
- (c) "Not For Resale (NFR) Version" means a version of the Software, so identified, to be used to review and evaluate the Software, only.*
- (d) "Software" means the VirusBuster Ltd. (R) VirusBuster(TM) software program supplied by VirusBuster Ltd. herewith, which may also include documentation, associated media, printed materials, and online and electronic documentation.*

### 2. License

*This EULA allows you to:*

- (a) Install and use the Software on a single computer; OR install and store the Software on a storage device, such as a network server, used only to run or install the Software on your other computers over an internal network, provided you have a license for each separate computer on which the Software is installed or run from the storage device. A license for the Software may not be shared or used concurrently on different computers.*
- (b) Educational and Home Version Only. If you have purchased a license for the Educational and/or the Home Version of the Software, then you may install or store the Software on a storage device, such as a network server, used only to run or install the Software on your other computers over an internal network for use by a total number of concurrent users not to exceed the number of user licenses you have been granted; provided, you agree to implement reasonable controls to ensure that your use of the Software does not exceed the number of licenses you have been granted. You agree that VirusBuster Ltd. may audit your use of the Software for compliance with the EULA at any time, upon reasonable notice.*
- (c) Make one copy of the Software in machine-readable form solely for backup purposes. You must reproduce on any such copy all copyright notices and any other proprietary legends on the original copy of the Software.*

### 3. License Restrictions

- (a) Other than as set forth in Section 2, you may not make or distribute copies of the Software, or electronically transfer the Software from one computer to another or over a network.*
- (b) You may not decompile, reverse engineer, disassemble, or otherwise reduce the Software to a human-perceivable form.*
- (c) You may not sell, rent, lease, transfer or sublicense the Software.*
- (d) You may not modify the Software or create derivative works based upon the Software.*
- (e) You may not use the Software in automatic, semi-automatic or manual tools designed to create virus signatures, virus detection routines, any other data or code for detecting malicious code or data.*
- (f) In the event that you fail to comply with this EULA, VirusBuster Ltd. may terminate the license and you must destroy all copies of the Software.*

### 4. Upgrades

*If this copy of the Software is an upgrade from an earlier version of the Software, it is provided to you on a license exchange basis. You agree by your installation and use of this copy of the Software to voluntarily terminate your earlier EULA and that you will not continue to use the earlier version of the Software or transfer it to another person or entity.*

### 5. Ownership

*The foregoing license gives you limited rights to use the Software. VirusBuster Ltd. and its suppliers retain all right, title and interest, including all copyrights, in and to the Software and all copies thereof. All rights not specifically granted in this EULA, including International Copyrights, are reserved by VirusBuster Ltd. and its suppliers.*

### 6. LIMITED WARRANTY AND DISCLAIMER

- (a) LIMITED WARRANTY. VirusBuster Ltd. warrants that, for a period of ninety (90) days from the date of delivery (as evidenced by a copy of your receipt) that the physical media on which the Software is furnished will be free from defects in*

materials and workmanship under normal use.

(b) NO OTHER WARRANTY. EXCEPT AS SET FORTH IN THE FOREGOING LIMITED WARRANTY, VirusBuster Ltd. AND ITS SUPPLIERS DISCLAIM ALL OTHER WARRANTIES, EITHER EXPRESS OR IMPLIED, OR OTHERWISE INCLUDING THE WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. ALSO, THERE IS NO WARRANTY OF NONINFRINGEMENT, TITLE OR QUIET ENJOYMENT. IF APPLICABLE LAW IMPLIES ANY WARRANTIES WITH RESPECT TO THE SOFTWARE, ALL SUCH WARRANTIES ARE LIMITED IN DURATION TO NINETY (90) DAYS FROM THE DATE OF DELIVERY. No verbal or written information or advice given by VirusBuster Ltd. its dealers, distributors, agents or employees shall create a warranty or in any way increase the scope of this warranty.

#### 7. Exclusive Remedy

Your exclusive remedy under Section 6 is to return the Software to the place you acquired it, with a copy of your receipt and a description of the problem. VirusBuster Ltd. will use reasonable commercial efforts to supply you with a replacement copy of the Software that substantially conforms to the documentation, provide a replacement for defective media. VirusBuster Ltd. shall have no responsibility if the Software has been altered in any way, if the media has been damaged by accident, abuse or misapplication, or if the failure arises out of use of the Software with other than a recommended hardware configuration.

#### 8. LIMITATION OF LIABILITY.

NEITHER VirusBuster Ltd. NOR ITS SUPPLIERS SHALL BE LIABLE TO YOU OR ANY THIRD PARTY FOR ANY INDIRECT, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES (INCLUDING DAMAGES FOR LOSS OF BUSINESS, LOSS OF PROFITS, BUSINESS INTERRUPTION OR THE LIKE), ARISING OUT OF THE USE OR INABILITY TO USE THE SOFTWARE OR THIS EULA BASED ON ANY THEORY OF LIABILITY INCLUDING BREACH OF CONTRACT, BREACH OF WARRANTY, TORT (INCLUDING NEGLIGENCE), PRODUCT LIABILITY OR OTHERWISE, EVEN IF VirusBuster Ltd. OR ITS REPRESENTATIVES HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES AND EVEN IF A REMEDY SET FORTH HEREIN IS FOUND TO HAVE FAILED OF ITS ESSENTIAL PURPOSE.

#### 9. Basis of Bargain

The Limited Warranty, Exclusive Remedies and Limited Liability set forth above are fundamental elements of the basis of the agreement between VirusBuster Ltd. and you. VirusBuster Ltd. would not be able to provide the Software on an economic basis without such limitations.

#### 10. Consumer End Users Only

The limitations or exclusions of warranties and liability contained in this EULA do not affect or prejudice the statutory rights of a consumer, i.e., a person acquiring goods otherwise than in the course of a business.

#### 11. General Provisions

The internal laws of Hungary shall govern this EULA. This EULA contains the complete agreement between the parties with respect to the subject matter hereof, and supersedes all prior or contemporaneous agreements or understandings, whether oral or written. All questions concerning this EULA shall be directed to VirusBuster Ltd.

VirusBuster and VirusBuster logo are trademarks or registered trademarks of VirusBuster Ltd. in Hungary and/or other countries. Other marks are the properties of their respective owners.

## CONTACT

This manual provides comprehensive information on operational of our virus protection product. If you have any additional questions about it or would like to share your experience or proposals with us do not hesitate to contact us! Turn to us with confidence, your demands and remarks will be respected.

Address VirusBuster Ltd.  
Budapest 1518,  
Pf. 54.  
Hungary

Phone (+36) 1 382-7000  
Fax (+36) 1 382-7007  
Web <http://www.virusbuster.hu>  
Support <https://support.virusbuster.hu>  
E-mail [sales@virusbuster.hu](mailto:sales@virusbuster.hu)  
[support@virusbuster.hu](mailto:support@virusbuster.hu)