

Vba32 Antivirus **User Guide**



VirusBlokAda

Copyright © 2005-2018 VirusBlokAda Ltd.

Documentation version: 1.66 (February 2018)

All rights reserved. All contents, graphics and texts, in this documentation are the property of VirusBlokAda Ltd. No part of this documentation may be reproduced in any form or by any means, including online and offline publications, without written permission from VirusBlokAda Ltd.

Microsoft® and Windows® are registered trademarks of Microsoft Corporation.

All other company and product names may be trademarks of the respective companies with which they are associated.

VirusBlokAda Ltd.

Smolenskaya Str., 15, 803 b

220088 Minsk, Belarus

Tel.: (+375 17) 294-84-29 – (sales department)

Tel.: (+375 17) 290-59-29 – (programming department)

WWW: www.anti-virus.by

E-mail: support-en@anti-virus.by

VirusBlokAda may make improvements or changes in the product described in this documentation at any time. The latest version of the documentation is available on the developer's web-site:

<ftp://anti-virus.by/pub/docs/english/>

Contents

1.	Introducing Vba32	5
1.1	Technical Support	5
1.2	Contacts	5
2.	Getting Started.....	6
2.1	Vba32 Components	6
2.2	Vba32 Products.....	6
2.3	Installing Vba32.....	7
2.4	Launching Vba32 First Time.....	8
3.	Program Interface.....	9
3.1	System Tray Menu	9
3.2	System Tray Icon.....	9
3.3	Vba32 Loader Main Window.....	10
3.3.1	General.....	10
3.3.2	Initialization.....	10
3.3.3	Additional.....	11
3.3.4	Devices	12
4.	Antivirus File Protection	13
4.1	On-Access File Scanning	13
4.1.1	Vba32 Monitor Overview.....	13
4.1.2	Vba32 Monitor Settings	13
4.1.3	Viewing Scanning Results	17
4.2	On-Demand File Scanning	17
4.2.1	Vba32 Scanner Overview.....	17
4.2.2	Configuring Vba32 Scanner.....	17
4.2.3	Scanning.....	22
4.2.4	Viewing Results of Scanning	23
4.2.5	Windows Explorer Context Menu	25
4.3	Windows/DOS Console Scanner	26
4.3.1	Using Vba32 Console Scanner.....	27
4.3.2	Command Line Keys	27
5.	Antivirus E-mail Protection	28
5.1	Vba32 Mail Filter.....	31
5.1.1	Configuring Vba32 Mail Filter	31
5.2	Vba32 Outlook Plug-in	33
5.2.1	Configuring Vba32 Outlook Plug-in	33
5.3	Vba32 TheBat! Plug-in	34
5.3.1	Configuring Vba32 TheBat! Plug-in	34
6.	Internet Antivirus Protection	35
6.1	Vba32 Script-Filter	35
6.1.1	Configuring Vba32 Script-Filter.....	35
6.1.2	Blocking Scripts.....	36
6.2	Vba32 Antidialer	36
6.2.1	Configuring Vba32 Antidialer.....	36
7.	Antivirus Quarantine	38
7.1	Vba32 Quarantine Overview	38
7.2	Vba32 Quarantine Main Window.....	38
7.3	Configuring Vba32 Quarantine	39
7.3.1	General.....	39
7.3.2	Maintenance	39
7.4	Working with Quarantined Files.....	40
7.4.1	Adding Files to Vba32 Quarantine	40
7.4.2	Scanning Quarantined Files.....	40
7.4.3	Removing Files from Vba32 Quarantine.....	41
7.4.4	Sending Quarantined Files for Detailed Analysis	41
7.4.5	Extracting Quarantined Files	42
7.4.6	Restoring Quarantined Files	42
8.	SendLogs Utility.....	43

8.1	Starting SendLogs	43
8.2	Welcome Screen	43
8.3	Collecting Report Files	43
8.4	Methods of Sending	43
8.4.1	Send Logs Directly	43
8.4.2	Send Logs by Mail Client	44
8.4.3	Saving Logs	44
9.	Updating Vba32	45
9.1	Update	45
9.1.1	Update Files	45
9.1.2	Automatic Update	45
9.1.3	Manual Update	46
9.2	Viewing Update Results	46
9.2.1	Update Window	46
9.2.2	Report File	46
10.	Vba32 Security	47
10.1	Using Password	47
10.2	Setting Password	47
10.3	Changing Password	47
10.4	Removing Password	48
11.	Activating Vba32	49
11.1	Launching Vba32 Activation	49
11.2	Using Key File	49
11.2.1	Browsing for Key File	49
11.3	Using Activation Code	50
11.3.1	Configuring Connection	51
11.3.2	Specifying User Information	51
11.3.3	Receiving Key File	52
11.4	Finishing Activation	53
12.	Vba32 Scheduler	54
12.1	Main and additional facilities of Vba32 Scheduler	54
12.1.1	Create task	54
12.1.2	Edit task	55
12.1.3	Delete task	55
12.1.4	Run task	55
12.1.5	Log View	56
12.1.6	Detailed task mapping	56
12.2	Action types	57
12.2.1	Process	57
12.2.2	Scanning	58
12.2.3	Update	61
12.3	Scheduling of launch time	61
12.3.1	Scheduling of launch time: Minutes	61
12.3.2	Scheduling of launch time: Hours	61
12.3.3	Scheduling of launch time: Days	62
12.3.4	Scheduling of launch time: Weeks	62
12.3.5	Scheduling of launch time: Month	62
12.3.6	Scheduling of launch time: Fixed date	63

1. Introducing Vba32

Vba32 Antivirus is a reliable and quick tool to detect and neutralize computer viruses, mail worms, trojans and other malware (backdoors, adware, spyware, etc). **Vba32** provides a high-performance protection of your personal computers, workstations and local network servers.

Vba32 protects your computer when you surf the Internet and use e-mail.

Vba32 has a powerful heuristic analyzer which protects against new viruses as well as their modifications.

Considerable base of known malicious programs and regular antivirus updates provide the most powerful protection of your computer. See also:

[Technical Support](#)
[Contacts](#)

1.1 Technical Support

The program is supported by company VirusBlokAda Ltd, operating on the basis of license License № 01019/5031714, issued by the Operational Analytical Center under the President of the Republic of Belarus. The license was extended on the basis of the order of November 6, 2009 № 79 for a period of five years, order № 91 of October 31, 2014 for a period of five years and is valid until December 13, 2019. In licensing the enterprise's right to carry out technical and (or) cryptographic protection of information.

Smolenskaya Str., 15, 803 b
220088 Minsk, Belarus

Tel.: (+375 17) 294-84-29 – (sales department)

Tel.: (+375 17) 290-59-29 – (programming department)

WWW: www.anti-virus.by

E-mail: support-en@anti-virus.by

1.2 Contacts

If you have questions about the program visit our resource <http://www.anti-virus.by/en/>, where you will get detailed information on types of technical support.

You can contact us via e-mail: support-en@anti-virus.by.

You can subscribe to our newsletter to receive the latest news of the company by e-mail.

Please send new viruses to: newvirus@anti-virus.by.

To make requesting for technical support more handy, you should use [Send Logs](#) utility.

2. Getting Started

The section describes how to implement antivirus protection of your computer quickly and effectively.

It contains the following:

[Vba32 Components](#)

[Vba32 Products](#)

[Installing Vba32](#)

[Launching Vba32 First Time](#)

2.1 Vba32 Components

Vba32 Antivirus is a set of components which provide reliable protection of your computer against malicious programs and computer viruses. Depending on the version you have purchased and the components you have installed the following modules can be used:

- **Vba32 Loader** is the main control panel of **Vba32**. It allows tracing the state of antivirus modules, launching Vba32 Scanner and modifying Vba32 Monitor settings. See also [Program Interface](#) to learn more about Vba32 Loader functions.
- **Vba32 Monitor** provides permanent protection of existed files as well as new ones obtained from network or other drives, downloaded from the Internet or received by e-mail. See [Active File Protection](#) to learn more about Vba32 Monitor functions.
- **Vba32 Scanner** allows performing on-demand antivirus scanning of drives, folders and files. It provides handy means to view scanning results. See [File Scanning on Demand](#) to learn more about Vba32 Scanner functions.
- **Windows/DOS Console Scanner** is designed for antivirus scanning of disks, folders and files using the command line. See [Windows/DOS Console Scanner](#) to learn more about its functions.
- **Windows Explorer context menu extension** allows scanning specified files from the Windows Explorer context menu. See [Windows Explorer Context Menu](#) to learn more about its functions.
- **Antivirus Mail Filter** scans e-mail messages received by POP3 or IMAP4 protocol before they are obtained by mail clients. See [Vba32 Mail Filter](#) to learn more about its functions.
- **Antivirus The Bat! Plug-in** is designed to protect The Bat! mail client, version 1.61 and creater. See [Vba32 The Bat Module](#) to learn more about its functions.
- **Antivirus Microsoft Outlook Plug-in** is designed to protect such mail clients as Microsoft Outlook and Microsoft Exchange Client. See [Vba32 Outlook Plug-in](#) to learn more about its functions.
- **Antivirus Script-Filter** protects Microsoft Internet Explorer, Microsoft Outlook Express and any other application which uses Microsoft Windows Scripting Host. See [Antivirus Internet Protection](#) to learn more about its functions.
- **Antidialer** - provides protection from unauthorized attempts to create connections to unknown phone numbers. See [Vba32 Antidialer](#) to learn more about its functions.
- **Antivirus Quarantine** - provides storage of suspicious and infected files which are placed there by Vba32 plug-ins. See [Antivirus Quarantine](#) to learn more about its functions.

2.2 Vba32 Products

Vba32 products consist of [Vba32 components](#) which were selected to provide most effective antivirus protection. Depending on the conditions you working in, you can choose from four **Vba32 products**, for:

- personal computers and workstation running Windows XP SP3;
- personal computers and workstation running Windows Vista/7/8/8.1/10;

- file servers running Windows Server 2003;
- file servers running Windows Server 2008/2008 R2/2012/2012 R2/2016.

Minimal system requirements:

Processor: Intel Pentium 4/AMD Athlon64;

Free RAM: 512 MB;

Free disk space: 512 MB.

Recommended system requirements:

Processor: Intel Core 2 Duo/AMD Athlon64 X2 or better;

Free RAM: 1 Gb or more;

Free disk space: 1 Gb or more.

2.3 Installing Vba32

Starting installation

Launch installation file that you received from your dealer or that you downloaded from the site of the developers.

Attention: Before starting the security installation, that you have administrator rights to the system.

Welcome screen

Press **Next** to continue installation. Press **View What's New** to learn about changes in this version of the product.

Selecting installation type

You can choose from three types of installation: Typical, Complete and Custom.

- **Typical** installation contains a standard set of components that provides effective antivirus protection of your computer. It is recommended for most users.
- **Complete** installation allows installing all Vba32 components. It is recommended for the best performance.
- **Custom** installation allows choosing Vba32 components to install. It is recommended for advanced users.

Press **Browse** to choose the destination folder for installation. Press **Next** to continue installation.

Selecting features

If the **Custom** installation type is selected list of the components is displayed in the next window. By default the number of components corresponds to the **Typical** installation ones. Press **Next** to continue installation.

Configuring Vba32

Vba32 can be configured in the next window:

- **Launch Vba32 Loader at Windows startup** - the program will be automatically loaded on the operating system startup.
- **Create Desktop Icons** - Vba32 Loader icon will be placed on the desktop.
- **Display Vba32 items in the Start menu** - Vba32 group will be created in the Start menu.

You should also specify the location of your Vba32 registration key file. Press the **Browse** button and specify the path to the vba32.key file in the dialog box. If you don't have the registration key contact your dealer.

Attention: The program will work in demo mode without registration key! You will be able to update the program only once and some features will be unavailable!


Ready to install the application

Press the **Next** button to start installing files. Press **Back** to change the previous settings or press **Cancel** to exit the setup program.

The installation has been successfully finished

Press the **Finish** button to complete the installation.


2.4 Launching Vba32 First Time

After installing Vba32 and rebooting the computer you will see Vba32 Loader icon  in the system tray. If there is no icon it means that you have disabled automatic Loader launching at Windows startup while installing; or you might have installed the program incorrectly (then it is recommended to reinstall it).

As the next step we recommend you to update the program to get the latest antivirus modules. You just have to right-click the Vba32 Loader system tray icon and select **Update** in the drop-down list. By default update is downloaded from <http://www.anti-virus.by/update/>. You can use alternate addresses from the program list or specify any other valid update path.

Notes: Hundreds of new viruses and their modifications are coming into being in the world every day. The annual damage caused by the viruses is estimated in billions. Therefore in time update of antivirus bases ensures reliable protection against computer viruses and malicious programs.

When update have been downloaded you can perform your first scanning. We recommend scanning all disks of your computer, because there is a chance that your computer was previously infected with malware or computer viruses to install anti-virus complex Vba32Select **Scanner** from the drop-down list or press the **Scanner** button in the main window of **Vba32 Loader**.

Then press  on the toolbar to start antivirus scanning. When the scanning is finished you will see the results in Vba32 Scanner window.

3. Program Interface

Vba32 interface is the main means of interaction with antivirus components. It provides control and visualization of actions in progress and simplifies usage and setting up of **Vba32 Loader**.

Vba32 Loader is a primary component of the package. It implements the graphical user interface. See also:

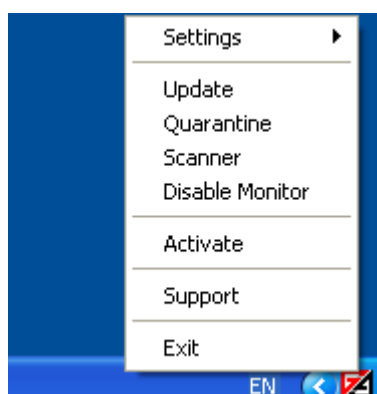
[System Tray Menu](#)

[System Tray Icon](#)

[Vba32 Loader Main Window](#)

3.1 System Tray Menu

The context menu of Vba32 Loader provides quick and handy access to main functions of the program.



Vba32 Loader context menu in system tray

Vba32 Loader menu includes the following items:








- **Settings** – invokes submenu of antivirus components.
 - **Loader** - invokes [Vba32 Loader Main Window](#).
 - **Monitor** - invokes [Vba32 Monitor settings](#).
 - ... - invokes settings of corresponding antivirus components.
- **Update** - launches automatic update of **Vba32**.
- **Scanner** – invokes the [Vba32 Scanner](#) main window.
- **Disable\Enable Monitor** - turns on\off [Vba32 Monitor](#).
- **Activate** - launches the [Vba32 activation utility](#). The item is displayed only if **Vba32** key file is missing or expired as well as there are less than 8 days till expiration.
- **Support** - invokes the dialog box with support information.
- **Exit** - unloads [Vba32 Loader](#) and exits the program.

3.2 System Tray Icon

System tray icon indicates the current state of antivirus protection and notifies a user about actions performed by the program. You can access main functions, settings and actions of **Vba32** with the help of the icon:

- Right-click the icon to invoke [system tray menu](#) and select a menu item.
- Double-click the icon to invoke [Vba32 Loader main window](#).


During Vba32 functioning the icon animation can be changed depending on the status of antivirus protection and actions performed by the program:

	- the protection of your computer is turned on , i.e. the kernel has been successfully loaded and Vba32 Monitor is enabled.
	- the protection of your computer is turned off , i.e. Vba32 Monitor is disabled.
	- files on disks are being scanned.
	- a mail message is being scanned.
	- the program is being updated.
	- the program has been updated or it is needed to confirm update start.
	- you have to reboot the computer to complete the update or an error occurred while updating.


3.3 Vba32 Loader Main Window

Vba32 Loader main window is your primary tool to work with the program. It contains various information about the program and provides access to its components.

If you want to invoke **Vba32 Loader** main window:

- Right-click  icon in the [system tray](#) to invoke [system tray menu](#). Select **Settings** and then **Loader**.

or

- Double-click  icon in the [system tray](#) to invoke the [Vba32 Loader Main Window](#).

Vba32 Loader main window has three tabs:

[General](#)

[Initialization](#)

[Additional](#)

[Devices](#)

3.3.1 General

General tab of the [Vba32 Loader main window](#) contains information about operating system version, user license, program version. Buttons in the window perform the following actions:

- **Password** - invokes **Change Password** dialog box. See [Vba32 Security](#) to get detailed information about changing password.
- **Monitor** - invokes [Vba32 Monitor](#) settings
- **Scanner** - invokes [Vba32 Scanner](#) settings
- **Exit** - disables antivirus protection and exits Vba32 Loader.

Attention: Turning off antivirus protection makes your computer vulnerable to malicious program and computer viruses. It isn't recommended disabling antivirus protection with no reason and for a long time.

3.3.2 Initialization

Initialization tab of the [Vba32 Loader main window](#) contains settings of the Vba32 Loader startup:

- **Launch Loader at Windows startup** - Vba32 Loader is launched automatically at startup of the operating system.
- **Enable Monitor at Loader startup** - Vba32 Monitor is enabled at Vba32 Loader startup.
- **Protect Loader process** - enables protection of Vba32 Loader process.
- **Display loading progress** - allows watching the process of Vba32 Loader loading.
- **Search for rootkits** - sets rootkit detection mode.

- **Scan memory** - Vba32 Loader scans memory of all processes, dynamic libraries and drivers at startup. There are three scanning modes:
 - Fast mode** - Vba32 Loader scans memory using fast algorithms, doesn't scan drivers and doesn't use heuristic analyzer. Moreover antivirus kernel cache is enabled, that speeds up further scanning of objects.
 - Full mode** - when working in this mode all objects in memory, including drivers, are scanned. Heuristic analyzer is enabled as well as antivirus kernel cache.
 - Excessive mode** - when working in this mode all objects in memory are scanned, antivirus kernel cache isn't used, heuristic level is excessive.
- **Scan boot sectors** - Vba32 Loader scans boot sectors at startup.
 - Scan floppy boot sectors** - Vba32 Loader scans floppy boot sectors at startup.
- **Scan files launched at system startup** - Vba32 Loader scans files launched at system startup.
- **Installation Folder** - displays the folder where Vba32 is installed to.

Press **Ok** to save any changes you have made and close the window.
 Press **Cancel** to close the window without saving.
 Press **Apply** to save any changes you have made without closing the window.
 Press **Help** to open Help file.


3.3.3 Additional

Additional tab of the [Vba32 Loader main window](#) contains settings of the Vba32 Loader functioning:

Report File Settings

- **Keep** - enables the mode when all actions performed by Vba32 Loader as well as their results are written to the report file. Press **Browse** to change report file name and path.
- **Add** - sets the mode of appending new information to the report file.
- **Maximum size, kB** - specifies maximum size of the report file. If this value is achieved, addition of records to the file end will cause deletion of records at the file beginning. It is recommended to enable it in order not to litter the system with stale data. To view the report file, press **Show**.

Interface settings

- **Sound Warning** - enables sound notification about crucial events (launching, Vba32 Monitor enabling and disabling, detection and cure of malicious programs, etc).
- **Tray icon animation** - displays animated [system tray](#) icon  when the Loader performs some actions.
- **Interface Language** - allows choosing the language from the drop-down list.

Update settings

- **Time intervals, hrs** - the program is updated automatically every period of time specified. Otherwise update should be started manually.
- **Interactive** - sets the interactive mode of automatic update.
- **Network Settings** - to change network settings, press **Settings**

Access to update resources - contains settings of authorization to access update resources.

Use proxy-server - specifies whether proxy-server is used to connect update server or not. Type **Address** and **Port** number of proxy-server.

Use this account to access update resource - enables authorization to access update resources. Specify **User name** and **Password**.

Update resources - lists of paths which will be used to update the program. Press **Add...** to add a new resource (URL, local folder or network (UNC) path). Press **Delete** to remove the selected resource from the list. Press **Edit...** to modify the resource selected in the list. Press **Up** to move the resource selected in the list up. Press **Down** to move the resource selected in the list down.

- **Path** - choose the path from the drop-down list or press **Settings** to specify the path manually. Press **Update** to start updating.
- **Last** - displays the date and time of the last successful update.
- **Next** - displays the date and time of the next update (if automatic update is enabled).

Press **Ok** to save any changes you have made and close the window.
Press **Cancel** to close the window without saving.
Press **Apply** to save any changes you have made without closing the window.
Press **Help** to open Help file.

3.3.4 Devices

Devices tab of the [Vba32 Loader main window](#) contains information about different devices: denied, authorized or unknown devices contains settings of the Vba32 Loader functioning:

Device security. This option allows configuring devices protection:

Disabled. All of USB-devices can be connected. The report isn't kept.

Enabled. Only authorized USB-devices can be connected. The report is kept.

Report. All of USB-devices can be connected but it's kept detailed report in which the name of the computer, connection time, dates of connection, action and serial number of USB-device are recorded.

Authorized devices. They are devices about which the user has accepted the decision and what have access to the computer.

Deny. The user refuses the connection of USB-device to the computer.

Delete. The user excludes devices from the list of authorized USB-devices.

Denied devices. They are devices about which the user has accepted the decision and what don't have access to the computer.

Allow. The USB-device is enabled to access to the computer.

Delete. The user excludes devices from the list of authorized USB-devices.

Unknown devices. USB-devices are connected but the user hasn't accepted the decision.

Allow. The USB-device is enabled to access to the computer.

Deny. The user refuses the connection of USB-device to the computer.

Delete. The user excludes devices from the list of authorized USB-devices.

Press **Ok** to save any changes you have made and close the window.
Press **Cancel** to close the window without saving.
Press **Apply** to save any changes you have made without closing the window.
Press **Help** to open Help file.

4. Antivirus File Protection

Antivirus file protection is one of the most important elements of the antivirus protection. Malicious programs and computer viruses corrupt user data and steal private information, so effective implementation of antivirus protection is a crucial point when preventing your data from being compromised.

See also:

[On-Access File Scanning](#)

On-Demand File Scanning

[Windows/DOS Console Scanner](#)

4.1 On-Access File Scanning

On-Access file scanning permanently protects your computer and immediately detects and cures viruses and malicious programs. Every time you open or write a new file to a disk the file will be scanned for viruses.

See also:

[Vba32 Monitor Overview](#)

[Vba32 Monitor Settings](#)

[Viewing Procession Results](#)

4.1.1 Vba32 Monitor Overview

Vba32 Monitor is a Vba32 component which performs permanent protection of your computer.

To open Vba32 Monitor:

- Invoke [Vba32 Loader main window](#).
- Press **Monitor** on the [General](#) tab.

4.1.2 Vba32 Monitor Settings

Vba32 Monitor settings allow you to specify objects to protect, actions performed on these objects by Vba32 Monitor, report file settings and to view statistics.

See also:

[General](#)

[Objects](#)

[Actions](#)

[Background Scanning](#)

[Report](#)

[Statistics](#)

4.1.2.1 General

General tab of the [Vba32 Monitor settings window](#) contains information about user license, program version. The **Disable\Enable** button turns off\on permanent antivirus protection.

Attention: Turning off antivirus protection makes your computer vulnerable to malicious program and computer viruses. It isn't recommended disabling antivirus protection with no reason and for a long time.

4.1.2.2 Objects

Objects tab contains the following settings:

- **Scan standard file types set** - Vba32 Monitor scans files with standard extensions such as
-

COM.EXE.DLL.DRV.SYS.OV?.VXD.SCR.CPL.OCX.PL.AX.PIF.DO?.XL?.
HLP.RTF.WI?.Z?.MSI.MSC.HT*.VB*.JS.JSE.ASP*.CGI.PHP*.?HTML.
BAT.CMD.EML.NWS.MSG.XML.MSO.WPS.PPT.PUB.JPG.JPEG.INF

- **Scan selected file types** - Vba32 Monitor scans files with extensions specified in the text field below and delimited by dots. Some file extensions are specified by default. Press **By default** to restore the initial list of extensions.
- **Scan all file types** - Vba32 Monitor scans all file types.

Excluding - allows specifying file extensions you want to exclude from scanning. These files may be ones that are not likely to be infected.

Note: Question mark (?) substitutes a single character in the extension. Asterisk (*) substitutes zero characters or a row of symbols of any length.

- **Scan only new files** - Vba32 Monitor scans **only** newly created and modified files of specified types. This reduces delays caused by Vba32 Monitor functioning, but diminishes protection reliability. Provided the mode is set, it is recommended to check the computer by [Vba32 Scanner](#) periodically.
- **Detect Spyware, Adware, Riskware** - Vba32 Monitor detects applications of Adware and Riskware types. They are considered as common infected files.
- **Folders and files excluded from Monitor scanning** - displays the list of files and folders that will **not** be scanned by Vba32 Monitor. To manage the list use the following buttons::
 - **Add** - to add a path to the list. Specify a path and a mask of files for exclusion (using '?' and '*' characters). You can also exclude files in the specified folders and subfolders.
 - **Edit** - to modify the path selected in the list of folders and files excluded from scanning by Vba32 Monitor.
 - **Delete** - to remove the selected path from the list.

Press **Ok** to save any changes you have made and close the window.
Press **Cancel** to close the window without saving.
Press **Apply** to save any changes you have made without closing the window.
Press **Help** to open Help file.

4.1.2.3 Background scanning

The **Background scanning** tab contains the settings of background scanning performed by Vba32 Monitor:

- **Scan files in background mode** - enables background scanning of files by Vba32 Monitor. The scanning is performed only if functioning conditions are complied with.
- **Functioning conditions** - allows determining conditions that should be complied with to perform the background scanning.
 - **Maximum CPU usage, %** - specifies maximum CPU usage in percents at which the background scanning is performed. CPU usage caused by Vba32 Loader is ignored.
 - **Maximum disk activity, %** - specifies maximum disk activity in percents at which the background scanning is performed.
 - **Maximum displacement of mouse pointer, px/s** - specifies maximum mouse pointer displacement at which Vba32 Monitor doesn't suspend the background scanning.
 - **Minimum battery charge, %** - specifies minimum battery charge in percents at which the background scanning is performed. If battery charge is less than the specified, the background scanning is suspended.
- **Scan files launched at system startup** - files launched at system startup are scanned when performing the background scanning.

- **Scan files frequently launched by user** - files frequently launched by a user are scanned when performing the background scanning.
- **List of paths to scan** - contains paths which are processed when performing the background scanning. The **Add**, **Edit** and **Delete** buttons can be used to manage the list.

Press **Ok** to save any changes you have made and close the window.
 Press **Cancel** to close the window without saving.
 Press **Apply** to save any changes you have made without closing the window.
 Press **Help** to open Help file.

4.1.2.4 Actions

Actions tab specifies actions performed by Vba32 Monitor when infected and suspicious files are detected.

- **Infected** - allows choosing an action from a drop-down list to perform on infected objects.

Block - an infected file will be blocked.

Cure - an infected file will be cured.

Delete - an infected file will be deleted.

Ask - every time Vba32 Monitor detects an infected file it will ask the user to choose an action.

Save copy to Quarantine - enables saving copies of infected files to [Vba32 Quarantine](#).

If previous action fails - allows choosing an action from the drop-down list to perform on infected objects if the action specified above fails.

Cure - an infected file will be cured.

Delete - an infected file will be deleted.

Ask - every time Vba32 Monitor detects an infected file it will ask the user to choose an action.

Save copy to Quarantine - enables saving copies of infected files to [Vba32 Quarantine](#).

If previous action fails - allows choosing an action from the drop-down list to perform on infected objects if actions specified in both lists above fail.

Delete - an infected file will be deleted.

Ask - every time Vba32 Monitor detects an infected file it will ask the user to choose an action.

Save copy to Quarantine - enables saving copies of infected files to [Vba32 Quarantine](#).

- **Heuristic Analysis** - allows detecting unknown malicious programs and modifications of known malicious programs. It provides a more reliable protection of your computer. You can choose the level of the heuristic analysis:

Disabled - unknown malicious programs will not be detected.

Optimal - practically doesn't slow down the scanning. Recommended for most of users.

Maximum - provides the maximum level of unknown malicious programs detection with the lowest probability of false positives; slows down scanning a bit.

Excessive - detects most of unknown malicious programs with the highest probability of false positives. Recommended for advanced users only.

Attention: Send suspicious files to newvirus@anti-virus.by for detailed analysis. This will help us in removing false positives in the next update of antivirus base.

- **Suspicious** - allows choosing an action from a drop-down list to perform on suspicious objects.
 - Skip** - a suspicious file will be skipped.
 - Block** - a suspicious file will be blocked.
 - Delete** - a suspicious file will be deleted.

Save copy to Quarantine - enables saving copies of suspicious files to [Vba32 Quarantine](#).

If previous action fails - allows choosing an action from the drop-down list to perform on suspicious objects if the action specified above fails.

- Skip** - a suspicious file will be skipped.
- Delete** - a suspicious file will be deleted.

Save copy to Quarantine - enables saving copies of suspicious files to [Vba32 Quarantine](#).

Press **Ok** to save any changes you have made and close the window.
 Press **Cancel** to close the window without saving.
 Press **Apply** to save any changes you have made without closing the window.
 Press **Help** to open Help file.

4.1.2.5 Report

Report tab contains parameters of keeping the report file.

- **Notify of Monitor actions** - sets the mode when Vba32 Monitor informs a user about its actions.
- **Report File**
 - **Keep** - sets the mode when all actions performed by Vba32 Monitor as well as their results are written to the report file. Default file is named `Vba32mNt.log` and placed in the Vba32 folder. Press **Browse** to specify another report file name and path.
 - **Add** - sets the mode of appending new information to the report file.
 - **Maximum size, kB** - specifies maximum size of the report file. If this value is achieved, addition of records to the file end will cause deletion of records at the file beginning. It is recommended to enable it in order not to litter the system with stale data. To view the report file, press **Show**.
 - **Information about "clean" files** - enables display of information about non-infected files in the report file.

4.1.2.6 Statistics

Statistics tab contains statistics of Vba32 Monitor functioning:

- General Monitor statistics:
 - **Scanned** - total number of scanned files.
 - **Suspicious** - number of detected suspicious files.
 - **Infected** - number of detected infected files.
 - **Blocked** - number of infected and suspicious files that were blocked.
 - **Cured** - number of cured files.
 - **Deleted** - number of removed files.
- **Last scanned file** - displays the last file scanned by Vba32 Monitor.
- **Last infected file** - displays the last file detected by Vba32 Monitor as infected.
- **Virus name** - name of the last malicious program or computer virus detected by Vba32 Monitor.

- **Statistics startup** - time and time when statistics was started. Press **Reset** to reset the counters and file names and to start new statistics.

4.1.3 Viewing Scanning Results

During Vba32 Monitor functioning statistics gathering is performed. The statistics provides the total number of scanned files, the number of detected infected and suspicious files as well as the number of actions performed. Moreover the statistics can be written to the report file.

See also:

[Scanning Statistics](#)

[Report File](#)

4.1.3.1 Scanning Statistics

To view scanning statistics:

- Invoke Vba32 Monitor window.
- Switch to the [Statistics](#) tab.

4.1.3.2 Report File

The report file contains detailed information about Vba32 Monitor functioning and its actions.

To view the report file:

- Invoke Vba32 Monitor window and switch to the [Report](#) tab.
- Press **Show** button.

See also [Vba32 Monitor Settings - Report](#).

4.2 On-Demand File Scanning

On-demand file scanning is an essential part of the high-performance antivirus protection of your computer.

See also:

[Vba32 Scanner Overview](#)

[Configuring Vba32 Scanner](#)

[Scanning](#)

[Viewing Results of Scanning](#)

[Windows Explorer Context Menu](#)

4.2.1 Vba32 Scanner Overview

Vba32 Scanner is a Vba32 component which performs scanning on user demand.

To launch Vba32 Scanner:

- Invoke [Vba32 Loader Main Window](#).
- Press **Scanner** button on the [General](#) tab.

The main window of Vba32 Scanner has a toolbar which provides handy means of working with scanning objects. See [Window Appearance](#) to learn more about the toolbar.

4.2.2 Configuring Vba32 Scanner

Vba32 Scanner settings allow you to change the appearance of Vba32 Scanner window, to choose disks, folders and files for scanning. Also, you can specify actions to perform by Vba32 Scanner and additional parameters of scanning.

See also:

[Window Appearance](#)

[Scanning Parameters](#)

[Configuration Files](#)

4.2.2.1 Window Appearance


You can decide by yourself how the Vba32 Scanner window should look like and what it should contain.

To change the window appearance, invoke **View** menu and select the following items:

- **Toolbar** - enables display of the toolbar.
- **Status Bar** - enables display of the status bar.
- **Folders** - enables tree view of folders on the left window panel.
- **Objects** - enables display of the list of files and folders on the right window pane.
- **Report** - enables display of scanning results within the lower part of the Vba32 Scanner window.
- **Large Icons** - makes icons large in the list of files and folders.
- **Small Icons** - makes icons small in the list of files and folders.
- **List** - arranges icons on the right panel as a list.

4.2.2.2 Scanning Parameters

Scanning parameters allow you to specify actions to perform by Vba32 Scanner on a file being scanned.

To change scanning parameters use  button on the Vba32 Scanner toolbar or invoke the **Settings** menu.

See also:

[Objects](#)

[Actions](#)

[Report](#)

[Additional](#)

4.2.2.2.1 Objects

Objects tab contains the following settings of Vba32 Scanner.

- **Scan standard file types set** - Vba32 Scanner processes files with standard extensions such as

```
COM.EXE.DLL.DRV.SYS.OV?.VXD.SCR.CPL.OCX.PL.AX.PIF.DO?.XL?.  
HLP.RTF.WI?.Z?.MSI.MSC.HT*.VB*.JS.JSE.ASP*.CGI.PHP*.*HTML.  
BAT.CMD.EML.NWS.MSG.XML.MSO.WPS.PPT.PUB.JPG.JPEG.INF
```
- **Scan selected file types** - Vba32 Scanner processes files with extensions specified in the text field below and delimited by dots. Some file extensions are specified by default. Press **By default** to restore the initial list of extensions.
- **Scan all file types** - Vba32 Scanner processes all file types.

Excluding - allows specifying file extensions you want to exclude from scanning. These files may be ones that are not likely to be infected.

Note: Question mark (?) substitutes a single character in the extension. Asterisk (*) substitutes zero characters or a row of symbols of any length.

- **Scan memory** - Vba32 Scanner checks memory as well as system processes and services at startup.
Fast mode - Vba32 Scanner processes only memory areas which are the most prone to be infected.
- **Scan boot sectors** - Vba32 Scanner processes boot sectors.

- **Scan files launched at system startup** - Vba32 Scanner processes files launched at system startup.
- **Scan mail** - Vba32 Scanner processes mail bases and messages.
- **Detect installers of malware** - Vba32 Scanner detects installers of malicious programs and computer viruses.
- **Scan archives** - Vba32 Scanner processes archived files.
 - **Maximum archive size, kB** - sets the limitation on maximum size of archived files being scanned.
- **Search for rootkits** - Vba32 Scanner checks the memory for rootkits.
- **Thorough mode** - sets the excessive mode of file scanning.

Attention: Thorough mode may considerably increase the time of file scanning.

- **Detect Spyware, Adware, Riskware** - Vba32 Scanner detects applications of Adware and Riskware types. They are considered as common infected files.

Press **Ok** to save any changes you have made and close the window.
Press **Cancel** to close the window without saving.

4.2.2.2.2 Actions

Actions tab specifies actions performed by Vba32 Scanner when infected and suspicious files are detected.

- **Infected files** - allows choosing an action from a drop-down list to perform on infected objects.
 - Skip** - an infected file will be skipped.
 - Cure** - an infected file will be cured.
 - Delete** - an infected file will be deleted.
 - Ask** - every time Vba32 Scanner detects an infected file it will ask the user to choose an action.

Save copy to Quarantine- enables saving copies of infected files to [Vba32 Quarantine](#).

If previous action fails - allows choosing an action from the drop-down list to perform on infected objects if the action specified above fails.

- Skip** - an infected file will be skipped
- Delete** - an infected file will be deleted.
- Ask** - every time Vba32 Scanner detects an infected file it will ask the user to choose an action.

Save copy to Quarantine - enables saving copies of infected files to [Vba32 Quarantine](#).

- **Archives containing viruses** - allows choosing an action from the drop-down list to perform on infected archives.
 - Skip** - the infected archive will be skipped.
 - Delete** - the infected archive will be removed.
 - Ask** - every time Vba32 Scanner detects an infected archive it will ask the user to choose an action.

Save copy to Quarantine - enables saving copies of infected archives to [Vba32 Quarantine](#).

- **Messages containing viruses** - allows choosing an action from the drop-down list to perform on infected messages.
 - Skip** - an infected mail message will be skipped.
 - Cure/Delete** - an infected mail message will be cured or deleted if cure fails.
 - Ask** - Vba32 Scanner will ask a user to choose an action.
- **Heuristic Analysis** - allows detecting unknown malicious programs and modifications of known malicious programs. It provides a more reliable protection of your computer. You can choose the level of the heuristic analysis:
 - Disabled** - unknown malicious programs will not be detected.
 - Optimal** - practically doesn't slow down the scanning. Recommended for most of users.
 - Maximum** - provides the maximum level of unknown malicious programs detection with the lowest probability of false positives; slows down scanning a bit.
 - Excessive** - detects most of unknown malicious programs with the highest probability of false positives. Recommended for advanced users only.

Attention: Send suspicious files to newvirus@anti-virus.by for detailed analysis. This will help us in removing false positives in the next update of antivirus base.

- **Suspicious** - allows choosing an action from a drop-down list to perform on suspicious objects.
 - Skip** - a suspicious file will be skipped.
 - Delete** - a suspicious file will be deleted.
 - Ask** - Vba32 Scanner will ask the user to choose an action.
 - Save copy to Quarantine** - enables saving copies of suspicious files to [Vba32 Quarantine](#).

Press **Ok** to save any changes you have made and close the window.
Press **Cancel** to close the window without saving.

4.2.2.2.3 Report

Report tab contains parameters of keeping the report file.

- **Display list of macros in documents** - enables display of all macros found during scanning MS Office documents. The following characters are used for substitution:

Character	Description
' '	unknown macro
'-'	empty macro
'='	known "good" macro
'!'	macro which is part of a known virus
'?'	damaged macro
'x'	removed macro
'C'	cured macro

- **Keep list of infected files** - Vba32 Scanner keeps a list of the infected files. Default file is `Vba32.lst` placed in the Vba32 folder. Press **Browse** to specify another file name and path. See [Scanning File List](#) to learn more about file lists.
- **Maximum lines in report window** - limits the maximum number of lines in report window. Default number is 128 lines. You can specify another number in the text field.
- **Report File**
 - **Keep** - sets the mode when all actions performed by Vba32 Scanner as well as their results are written to the report file. Default file is `Vba32Gui.log` placed in the Vba32 folder. Press **Browse** to specify another report file name and path.

- **Add** - sets the mode of appending new information to the report file.
- **Maximum size, kB** - specifies maximum size of the report file. If this value is achieved, addition of records to the file end will cause deletion of records at the file beginning. It is recommended to enable it in order not to litter the system with stale data. To view the report file, press **Show**.
- **Information about "clean" files** - enables display of information about non-infected files in the report file.

Press **Ok** to save any changes you have made and close the window.
Press **Cancel** to close the window without saving.

4.2.2.2.4 Additional

Additional tab contains additional parameters of Vba32 Scanner.

- **Configuration file name** - allows you to specify the name of the configuration file where Vba32 Scanner settings and selected scanning folders will be saved to. Default file is named `Vba32Gui.cfg` and placed in Vba32 folder. Press **Browse** to specify another configuration file name and path.
- **Load configuration file at startup** - configuration file is loaded at the Vba32 Scanner startup.
- **Save configuration file at exit** - configuration file is saved when exiting Vba32 Scanner.

Note: See [Configuration Files](#) to get detailed information about configuration files.

- **Enable cache when scanning objects** - Vba32 Scanner processes modified files only. This considerably diminishes the scanning time.
- **Priority of scanning** - allows you to change the priority of scanning. The higher it is the faster the scanning is but the lower the performance of other applications is.

Press **Ok** to save any changes you have made and close the window.
Press **Cancel** to close the window without saving.

4.2.2.3 Configuration Files

Vba32 configuration files usually have the *.cfg extension and are used to store settings of scanning. Several variants of settings can be created and used every time doing scanning.

See also:

[Saving Configuration File](#)

[Loading Configuration File](#)

4.2.2.3.1 Saving Configuration File

You can save current Vba32 Scanner settings in the configuration file for further usage. See more in [Loading Configuration File](#).

To save configuration file, invoke the menu **File** and select **Save Configuration File**. In the dialogue box specify the file name.

Also, you can enable automatic saving of the current settings to the configuration file when you close the Scanner. For this:

Invoke the [Settings](#) dialog box.

Switch to the [Additional](#) tab.

Specify the configuration file name (default file is named `Vba32Gui.cfg` and placed in the Vba32 folder).

Set Save configuration file at exit option.

4.2.2.3.2 Loading Configuration File

You can easily change Vba32 Scanner settings by loading corresponding configuration files.

To load the configuration file, invoke the **File** menu and choose **Load Configuration File**. Specify the name of configuration file to load in the dialog box.

Also, you can enable automatic loading of configuration file at the Vba32 Scanner startup. For this:

Invoke the [Settings](#) dialog box.

Switch to the [Additional](#) tab.

Specify configuration file name (default file is named Vba32Gui.cfg and placed in the Vba32 folder).

Set the Load configuration file at startup option.

4.2.3 Scanning

You can start, stop, pause and resume scanning when working with Vba32 Scanner.

See also:

[Choosing Objects for Scanning](#)

[Start Scanning](#)



[Pause and Resume Scanning](#)

[Stop Scanning](#)

[Scanning File List](#)

4.2.3.1 Choosing Objects for Scanning

Vba32 Scanner allows you to process local and network disks, files and folders. If you want to scan an object you have to add it to the scan list. You can form the list either in the directory tree or in the object panel. Selected objects are marked with the following signs:

-  - indicates that if the selected object contains folders or files they will be scanned as well.
-  - indicates that some enclosed objects have been removed from the scanning list.

To add a disk, folder or file to the scanning list:

- Hold **Ctrl** and click on the selected object. Another click will remove the object from the list.

or

- Select the object and press '**Space**' or '**+**'.

To remove the object from the scan list:

- Hold **Ctrl** and click on the selected object.


or

- Select the object and press '**Space**' or '**+**'.

4.2.3.2 Start Scanning

After [scanning objects](#) and [scanning parameters](#) have been defined or the corresponding [configuration file](#) has been loaded, you can start scanning.

To do this:

- Press the  button on the toolbar of the Vba32 Scanner main window.


or

- Invoke the **Scanning** menu and choose **Start Scanning**.

4.2.3.3 Pause and Resume Scanning

You can pause and resume scanning when working with Vba32 Scanner.


To pause scanning:

- Press the  button on the toolbar of the Vba32 Scanner main window.

or

- Invoke the **Scanning** menu and choose **Pause Scanning**.

To resume scanning:


- Press the  button on the toolbar of the Vba32 Scanner main window again.

or

- Invoke the **Scanning** menu and choose **Continue Scanning**.

4.2.3.4 Stop Scanning

To stop scanning:

- Press the  button on the toolbar of the Vba32 Scanner main window.

or

- Invoke the **Scanning** and choose **Stop Scanning**.

4.2.3.5 Scanning File List

Vba32 file lists have the *.lst extension and scanning the files they contain. By default Vba32 Scanner uses file list for infected files only (see more in [Vba32 Scanner Settings - Scanning Parameters - Report](#)). However you can form your own file list.

To scan file list:

Invoke the **Scanning** menu and choose **Scan a List**.

Specify your own file list in the dialog box.



Then the Scanner will start checking the files enumerated in the list. You can [pause](#), [resume](#) and [stop](#) scanning during the process.

4.2.3.6 Actions after Scanning

You can specify an action which is performed after scanning has been finished. Depending on a computer one of the following can be used:

- **Do Nothing** - no action is performed after scanning has been finished
- **Shut down** - the computer is shut down after scanning has been finished
- **Stand by** - the computer is switched to the Stand by mode after scanning has been finished
- **Hibernate** - the computer is switched to the Hibernate mode after scanning has been finished

To define the action:

- Press the  button on the toolbar of the Vba32 Scanner main window and choose from a menu item. If the action has been defined the button looks like .

4.2.4 Viewing Results of Scanning

Results of each scanning are displayed in the Vba32 Scanner result pane as well as in the report file.

See also:

[Result Pane](#)

[Report File](#)

4.2.4.1 Result Pane

Result Pane is a pane in the lowest part of the Vba32 Scanner main window. It allows watching the process of scanning, getting scanning results as a tree of objects or as a report and viewing scanning statistics.

See also:

[Object](#)

[Statistics](#)

[Report](#)

4.2.4.1.1 Objects

Objects tab allows watching the process of scanning and getting scanning results as a tree of objects. Depending on the [Vba32 Scanner settings](#) the following elements can be the root ones:

- Processes
- Boot sectors
- Autoruns
- File system

Information about tree elements is displayed as a table with the following columns:

- **File name** - name of a tree element. It can be a process name, a boot sector id, a path to autorun file or a path to infected or suspicious file.
- **File size, bytes** - size of file objects.
- **State** - state of scanning object.
- **Information** - information about a virus or a suspicion.
- **Action** - action has been performed on an object or, if it fails, reason of failure.
- **Copy** - takes the *Yes* value, if an object has been placed to Vba32 Quarantine, *No* otherwise.

Depending on its type some action can be performed on an object. Right click the object to invoke a context menu with the following items:

- **Show file** - opens the location of the object in Windows Explorer.
- **Cure** - cures the object.
- **Delete** - deletes the object.
- **Copy to Quarantine** - places a copy of the object to Vba32 Quarantine.
- **Expand all** - if the object is a root one and collapsed, it will be expanded and show all its child elements.
- **Collapse all** - if the object is a root one and expanded, it will be collapsed and hide all its child elements.
- **Select all** - selects all objects in the table.
- **Copy text** - copies the line containing the object to the clipboard.
- **Terminate process** - if the object is a process it will be terminated.

During scanning the **Objects** tab displays type of processing object and its status. After scanning has been finished the tab contains information about scanned object of the **Processes**, **Autorun** and **Boot sectors** root elements. Records about objects of **File system** is shown only if they are infected or suspicious.

4.2.4.1.2 Statistics

Statistics tab contains statistics of last scanning:

- **Scanning object** - name of an object being scanned at the moment.
- **Processes** - result of scanning processes running in the system.
- **Boot sectors** - result of scanning boot sectors.

- **Autorun** - result of scanning autorun.
- **File system** - result of scanning file system.
- **Found unique malicious programs** - number of unique malicious programs found during the scanning.
- **Found unique suspicions** - number of unique suspicious objects found during the scanning.
- **Total time** - total time of the scanning.
- **Files on disks** - statistics regarding scanning of disk files:
 - **scanned** - number of scanned files on disks.
 - **suspicious** - number of suspicious files on disks.
 - **infected** - number of infected files on disks.
 - **copies created** - number of copies created.
 - **cured** - number of cured files.
 - **deleted** - number of deleted files.

4.2.4.1.3 Report

Report tab contains a fragment of the [report file](#) with lines regarding the last scanning.

4.2.4.2 Report File

The report file contains detailed information about Vba32 Scanner functioning.

To view the report file:

- Press the  button on the toolbar of the Vba32 Scanner main window.

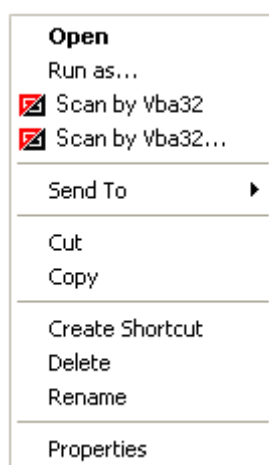
or

- Invoke the **File** menu and choose **View Report**.

See [Configuring Vba32 Scanner](#) - [Scanning Parameters](#) - [Report](#) to get detailed information about report file.

4.2.5 Windows Explorer Context Menu

Windows Explorer context menu items allow scanning files and folders without launching Vba32 Scanner.



Windows Explorer Context Menu

See also:

[Scanning](#)

[Scanning with Parameters](#)

[Results of Scanning](#)

4.2.5.1 Scanning

To start scanning with the help of Windows Explorer Context Menu, right-click an object and choose **Scan by Vba32**. Scanning process is shown in the [scanning result window](#).

Vba32 Scanner will perform scanning with default settings. See also [Scanning with Parameters](#) to get more information about customizing settings.

4.2.5.2 Scanning with Parameters

To configure scanning parameters and scan an object, right-click it and choose **Scan by Vba32...** The **Scanning Settings** dialog box contains the following settings:

- **Infected** - allows choosing an action from the drop-down list on infected objects:
 - Skip** - an infected file will be skipped.
 - Cure** - an infected file will be cured.
 - Delete** - the infected file will be deleted.
- **If Cure fails** - allows choosing an action from the drop-down list to perform on infected objects if Cure fails:
 - Skip** - an infected file will be skipped
 - Delete** - an infected file will be removed.
- **Thorough mode** - sets the excessive mode of file scanning.

Attention: Thorough mode may considerably increase the time of file scanning.

- **Scan mail** - Vba32 scans mail bases and messages.
- **Detect installers of malware** - Vba32 detects installers of malicious programs and computer viruses.
- **Scan archives** - Vba32 scans archived files.
- **Show macros in documents** - enables display of all macros found during scanning MS Office documents.
- **Heuristic Analysis** - allows detecting unknown malicious programs and modifications of known malicious programs. It provides a more reliable protection of your computer. You can choose the level of the heuristic analysis:
 - Disabled** - unknown malicious programs will not be detected.
 - Optimal** - practically doesn't slow down the scanning. Recommended for most of users.
 - Maximum** - provides the maximum level of unknown malicious programs detection with the lowest probability of false positives; slows down scanning a bit.
 - Excessive** - detects most of unknown malicious programs with the highest probability of false positives. Recommended for advanced users only.

Press **Scan** to start scanning.
Press **Cancel** to close the dialog box without scanning.
Press **Save Settings** to set the current settings as default ones.

4.2.5.3 Results of Scanning

After scanning has been started **Results of Scanning** window is invoked. It shows which objects are being scanned as well as their state. To terminate scanning, press **Terminate**.

When scanning has been finished press **Close** to close the window.

4.3 Windows/Linux Console Scanner

Windows/DOS Console Scanner is designed to perform antivirus scanning using the command line.

See also:

[Using Vba32 Console Scanner](#)

[Command Line Keys](#)

4.3.1 Using Vba32 Console Scanner

Make the folder, where Vba32 is installed to, the current one, to start Vba32 Console Scanner. The default destination folder is `c:\Program Files\Vba32\`

Command line syntax:

For Windows:

```
vba32w.exe [path] ... [path] [/key] ... [/key]
```

For DOS:

```
vba32x.exe [path] ... [path] [/key] ... [/key]
```

Command line syntax requires strict order: write paths, only then keys.

Path	meaning
file/folder	Path to a file or a folder to scan. Long file names should be in quotes.
*:	All local disks.
**:	All network disks.
@list	File list.

The `key` parameter sets operating modes of the program. See [Command Line Keys](#).

Note: Default parameters are: `/QU /MR /BT /AS /RW`

Press `ctrl+c` to finish Vba32 Console Scanner work.

4.3.2 Command Line Keys

The following table shows all command line keys used when [using Vba32 Console Scanner](#) for Linux.

Key	Description
/?[+ -]	display help screen;
H[+ -]	display help screen;
HELP[+ -]	display help screen;
/M=1	fast scan mode;
/M=2	optimal scan mode (/AF+);
/M=3	thorough scan mode (/AF+ /PM+);
/AF[+ -]	all files;
-SL[+ -]	follow symbolic link;
/PM[+ -]	thorough scan mode;
/RW[+ -]	detect Spyware, Adware, Riskware;
/CH[+ -]	turn on cache while scanning object;
/FC[+ -]	cure virus-infected files;
/FD[+ -]	delete virus-infected files;
/FR[+ -]	rename virus-infected files;
/FM+[directory]	move infected files to the selected directory;

SD[+ -]	delete suspect files;
SR[+ -]	rename suspect files;
SM+[directory]	move suspect files to the selected directory (by default /var/virus);
/HA=[0 1 2 3]	heuristic analysis level (0 - disabled, 2 - maximum);
/QI+[directory] -]	copy infected object to Quarantine;
/QS+[directory] -]	copy suspicious object to Quarantine;
/D=[N,][file name]	run program once in N days (1 by default);
/R=[file name]	saving report to a file (VBA32.RPT by default);
/R+[[file name]	append report to a file (VBA32.RPT by default);
/L=[file name]	save the list of infected files to a file(VBA32.LST);
/L+[имя_файла]	append the list of infected files to a file (VBA32.LST);
/QU[+ -]	allow program termination (by default);
/OK[+ -]	include "clean" file names in the report;
/AR[+ -]	scan archives;
/AD[+ -]	delete archives containing infected files;
/AL=[file_size,kB]	don't scan archives larger than the specified value;
/SFX[+ -]	scan installers of malware;
/ML[+ -]	scan mail;
/MD[+ -]	delete messages containing infected files;
/VL[+ -]	view the list of viruses known to the program;
/VM[+ -]	display information about macros in documents;
/SI[+ -]	additional information about program support;
/LNG=suffix	use language support file VBA32.LNG (RU by default);
/KF={directory path}	specify path to a key file;
/EXT=	specify the list of file extensions to be scanned;
/EXT+	add user defined file extensions to the default list;
/EXT-	remove file extensions from the default list;
/WK[+ -]	wait for any key to be pressed when finished.

These options are enabled by default: -QU-RW

The following table shows all command line keys used when [using Vba32 Console Scanner](#) for Windows.

Key	Description
/?[+ -]	display help screen;
H[+ -]	display help screen;
HELP[+ -]	display help screen;
/M=1	fast scan mode;
/M=2	optimal scan mode (/AF+);
/M=3	thorough scan mode (/AF+ /PM+);
/AF[+ -]	all files;

/PM[+ -]	thorough scan mode;
/RW[+ -]	detect Spyware, Adware, Riskware;
/CH[+ -]	turn on cache while scanning object;
/FC[+ -]	cure virus-infected files;
/FD[+ -]	delete virus-infected files;
/FR[+ -]	rename virus-infected files;
/FM+[directory]	move infected files to the selected directory;
SD[+ -]	delete suspect files;
SR[+ -]	rename suspect files;
SM+[directory]	move suspect files to the selected directory (by default /var/virus);
/BC[+ -]	cure boot sectors;
/NA[+ -]	disable signed files detection (only Windows);
/LF[+ -]	load Russian font (DOS-version only);
/HA=[0 1 2 3]	heuristic analysis level (0 - disabled, 2 - maximum);
/MR=[0 1 2]	scan your memory (0 - disabled, 2 - full, full mode is enabled by default (only Windows));
/AS=[0 1 2]	scan autorun files (0 - disabled, 2 - full, full mode is enabled by default (only Windows));
/BT[+ -]	scan boot sectors (enabled by default)
/QI+[directory] -]	copy infected object to Quarantine;
/QS+[directory] -]	copy suspicious object to Quarantine;
/D=[N,][file name]	run program once in N days (1 by default);
/R=[file name]	saving report to a file (VBA32.RPT by default);
/R+[[file name]	append report to a file (VBA32.RPT by default);
/UL[+ -]	UTF-8 log file;
/L=[file name]	save the list of infected files to a file(VBA32.LST);
/L+[имя_файла]	append the list of infected files to a file (VBA32.LST);
/QU[+ -]	allow program termination (by default);
/DB=[directory]	search during databases updating in specified directory;
/SS[+ -]	enable audio alarm when detecting virus;
/OK[+ -]	include "clean" file names in the report;
/AR[+ -]	scan archives;
/AD[+ -]	delete archives containing infected files;
/AL=[file_size,kB]	don't scan archives larger than the specified value;
/SFX[+ -]	scan installers of malware;
/ML[+ -]	scan mail;
/MD[+ -]	delete messages containing infected files;
/VL[+ -]	view the list of viruses known to the program;
/VM[+ -]	display information about macros in documents;
/SI[+ -]	additional information about program support;

/LNG=suffix	use language support file VBA32.LNG (RU by default);
/KF={directory path}	specify path to a key file;
/EXT=	specify the list of file extensions to be scanned;
/EXT+	add user defined file extensions to the default list;
/EXT-	remove file extensions from the default list;
/WK[+ -]	wait for any key to be pressed when finished.

These options are enabled by default: /QU /MR /BT /AS /RW

5. Antivirus E-mail Protection

Because of its popularity the Internet and e-mail are the main means for malware to spread all over the world. Many malicious programs, so-called 'worms', send their copies as attachments of e-mail messages. For that, they use addresses from user address books.

Vba32 antivirus components provide reliable protection of your computer against malicious programs and computer viruses spreading via e-mail systems.

See also:

[Vba32 Mail Filter](#)

[Vba32 Outlook Plug-in](#)

[Vba32 TheBat! Plug-in](#)


5.1 Vba32 Mail Filter

Vba32 Mail Filter is a Vba32 component which provides antivirus protection of any mail clients that use POP3 and IMAP protocols (Outlook Express, The Bat!, MS Outlook, etc).

5.1.1 Configuring Vba32 Mail Filter

Vba32 Mail Filter settings allow specifying actions that it will perform when scanning e-mail, defining from which servers messages are intercepted, configuring parameters of keeping the report file and viewing statistics.

To invoke the **Vba32 Mail Filter** Settings dialog box:

- Right-click the  [tray icon](#), to invoke the [system tray menu](#).
- Select **Settings**.
- Then select **Mail Filter**.

The **Vba32 Mail Filter** dialog box contains the following tabs:

[Objects](#)

[Interception](#)

[Statistics](#)

5.1.1.1 Objects

Objects tab contains settings of actions performed by Vba32 Mail Filter if infected and suspicious attachments are detected.

- **Scan mail** - enables Vba32 Mail Filter. Provides protection of mail regardless of mail clients (Outlook Express, The Bat!, MS Outlook, etc).
- **Infected Messages** - allows choosing an action from the drop-down list performed on messages with infected attachments:
 - **Receive** - the infected message will be received without any actions undertaken.
 - **Cure** - the infected message will be cured.
 - **Delete** - the infected message will be removed.

Save copy to Quarantine - an infected message will be copied to Vba32 Quarantine.

- Suspicious

Heuristic Analysis - allows detecting unknown malicious programs and modifications of known malicious programs. It provides a more reliable protection of your computer. You can choose the level of the heuristic analysis:

- **Disabled** - unknown malicious programs will not be detected.
- **Optimal** - practically doesn't slow down the scanning. Recommended for most of users.

- **Maximum** - provides the maximum level of unknown malicious programs detection with the lowest probability of false positives; slows down scanning a bit.
- **Excessive** - detects most of unknown malicious programs with the highest probability of false positives. Recommended for advanced users only.

Attention: Send suspicious files to newvirus@anti-virus.by for detailed analysis. This will help us in removing false positives in the next update of antivirus base.

Messages - allows choosing an action from the drop-down list performed on suspicious messages.

- **Receive** - a suspicious message will be received with no actions performed.
- **Delete** - a suspicious message will be deleted.

Save copy to Quarantine - a suspicious message will be copied to Vba32 Quarantine.

- **Messages being deleted are replaced with:** - a dialog box which allows forming message template that deleted messages will be replaced with is invoked by pressing **Template...**

5.1.1.2 Interception

Interception tab contains the list of servers from which the messages will be intercepted.

- **Port** - specifies server port from which messages will be intercepted (default port numbers are 110 (POP3) and 143 (IMAP)).
- **Server** - specifies server name or IP-address from which messages will be intercepted (* stands for any server).

Press **Add**, **Change** or **Delete** to manage the list.

5.1.1.3 Statistics

Statistics tab displays statistics of Vba32 Mail Filter functioning and allows configuring parameters of keeping the report file:

- **Scanned messages:** - number of mail messages scanned by Vba32 Mail Filter. Press **Reset** to reset the statistics and to start the new one.
- **Infected** - statistics of scanned mail messages which contain attachments with malicious programs or scripts.
 - **Total** - total number of infected mail messages detected by Vba32 Mail Filter.
 - **Cured** - number of cured mail messages.
 - **Deleted** - number of deleted infected mail messages.
 - **Quarantined** - number of infected mail messages placed to Vba32 Quarantine.
 - **Virus name** - name of the last malicious object detected.
 - **Time** - date and time of the last malicious object detection.
- **Suspicious** - statistics of scanned mail messages which contain attachments with features of malicious programs or with modifications of known malicious programs.
 - **Total** - total number of suspicious mail messages detected by Vba32 Mail Filter.
 - **Deleted** - number of deleted suspicious mail messages.
 - **Quarantined** - number of suspicious mail messages placed to Vba32 Quarantine.
 - **Suspected of being**- heuristic analyzer information about the last detected suspicious object.
 - **Time** - date and time of the last suspicious object detection.
- **Report File**
 - **Keep** - sets the mode when all actions performed by Vba32 Mail Filter as well as their results are written to the report file. Default file is named `vba32pp3.log` and placed in the program folder. Press **Browse** to specify another name and path.

- **Detailed** - sets the mode of appending new information to the report file.
- **Maximum size, kB** - specifies maximum size of the report file. If this value is achieved, addition of records to the file end will cause deletion of records at the file beginning. It is recommended to enable it in order not to litter the system with stale data. To view the report file, press **Show**.

5.2 Vba32 Outlook Plug-in

Vba32 Outlook Plug-in is a Vba32 component which protects Microsoft Outlook® mail client against viruses and malicious programs.

Vba32 Outlook Plug-in detects malicious attachments in mail messages and performs actions on them specified by a user. It provides protection of your computer from malicious programs sending via e-mail. Moreover Vba32 Outlook Plug-in scans outgoing mail preventing further spread of malicious programs.

5.2.1 Configuring Vba32 Outlook Plug-in

To invoke the **Vba32 Outlook Plug-in** settings dialog box:

- Right-click [tray](#) icon  to invoke [system tray menu](#).
- Select **Settings**.
- Then select **Outlook Plug-in**.

The **Vba32 Outlook Plug-in** dialog box contains the following settings:

- **General**
 - **Scan messages before reading** - Vba32 Outlook Plug-in scans received messages before reading.
 - **Scan messages before sending** - Vba32 Outlook Plug-in scans messages before sending.
 - **Notify of actions** - Vba32 Outlook Plug-in notify a user about all detected malicious and suspicious attachments as well as actions performed on them.
- **Infected objects**

Action 1 - allows choosing an action from the drop-down list to perform on messages with infected attachments.

 - **Cure** - an infected message will be cured.
 - **Delete** - an infected message will be deleted.
 - **Ask** - every time Vba32 Outlook Plug-in detects an infected attachment it will ask a user to choose an action.

Save copy to Quarantine - an infected message will be copied to Vba32 Quarantine.

Replace object with... - press it to view and modify message template which will replace infected messages being deleted.

Action 2 - allows choosing an action from the drop-down list to perform on infected attachments if previous action fails.

 - **Delete** - the infected message will be removed from the system.
 - **Skip** - the infected message will be skipped without any actions performed on it.
 - **Ask** - every time Vba32 Outlook Plug-in detects an infected message it will ask the user to choose an action.

Save copy to Quarantine - an infected message will be copied to Vba32 Quarantine.
- **Suspicious objects**

Heuristic Analysis - allows detecting unknown malicious programs and modifications of known malicious programs. It provides a more reliable protection of your computer.

You can choose the level of the heuristic analysis:

- **Disabled** - unknown malicious programs will not be detected.
- **Optimal** - practically doesn't slow down the scanning. Recommended for most of users.
- **Maximum** - provides the maximum level of unknown malicious programs detection with the lowest probability of false positives; slows down scanning a bit.
- **Excessive** - detects most of unknown malicious programs with the highest probability of false positives. Recommended for advanced users only.

Attention: Send suspicious files to newvirus@anti-virus.by for detailed analysis. This will help us in removing false positives in the next update of antivirus base.

Action - allows choosing an action from the drop-down list to perform on messages with suspicious attachments.

- **Delete** - a suspicious message will be deleted.
- **Skip** - a suspicious message will be skipped.
- **Ask** - every time Vba32 Outlook Plug-in detects an suspicious attachment it will ask a user to choose an action.

Save copy to Quarantine - a suspicious message will be copied to Vba32 Quarantine.

Replace object with... - press it to view and modify message template which will replace suspicious messages being deleted.

Press **Ok** to save any changes you have made and close the window.
Press **Cancel** to close the window without saving.
Press **Apply** to save any changes you have made without closing the window.

5.3 Vba32 TheBat! Plug-in

Vba32 TheBat! Plug-in is a Vba32 component which protects The Bat!® mail client against viruses and malicious programs.

Vba32 TheBat! Plug-in detects malicious attachments in mail messages and performs actions on them specified by a user. It provides protection of your computer from malicious programs sending via e-mail. Moreover Vba32 TheBat! Plug-in scans outgoing mail preventing further spread of malicious programs.

5.3.1 Configuring Vba32 TheBat! Plug-in

To get detailed information about configuring **Vba32 TheBat! Plug-in**, see The Bat! Help file, "Antivirus Protection".

6. Internet Antivirus Protection

Using the Internet for work, search or communication, you run risks to be infected by malicious programs. Many viruses, trojans and scripts can be embedded into HTML - pages. When visiting web-sites with such pages, you launch malware automatically and it may damage your computer..

[Vba32 Script-Filter](#) is a Vba32 component which is designed to protect your computer during working in the Internet. It intercepts and blocks malicious scripts embedded into HTML - pages.

There is one more threat you can face when working in the Internet - malicious dialers. A dialer is a computer program which creates a connection to the Internet or another computer network over the analog telephone or ISDN network. If it is installed at your computer, a malicious dialer can either replace a telephone number used to connect your provider via modem or hiddenly create a connection to an unknown Internet provider. As a rule, an unknown Internet provider is situated somewhere in exotic country and its services are much more expensive than services of your provider.

[Vba32 Antidialer](#) is a Vba32 component which is designed to intercept and block unauthorized attempts to create connections to unknown phone numbers.

See also:

[Vba32 Script-Filter](#)

[Vba32 Antidialer](#)

6.1 Vba32 Script-Filter


Vba32 Script-Filter provides antivirus protection of Microsoft Internet Explorer, Microsoft Outlook Express and any other applications that use Microsoft Windows Script Host (WSH).

Vba32 Script-Filter intercepts active scripts and scans them for viruses. If Vba32 Script-Filter detects a malicious script its execution is blocked.

Note: Since 3.10.3 version Vba32 Script-Filter is part of standard installation packages of Vba32 Workstation and Vba32 Personal (see [Vba32 Products](#) for details).

6.1.1 Configuring Vba32 Script-Filter

To invoke the **Vba32 Script-Filter** dialog box:

- Right-click the  [tray icon](#) to invoke [system tray menu](#).
- Point to the **Settings**.
- Then choose **Script-Filter**.

The **Vba32 Script-Filter** dialog box contains the following settings:

- **Enable Script-Filter** - turns on protection against malicious scripts executed by Microsoft Internet Explorer and Microsoft Outlook Express or any other application using Microsoft Windows Scripting Host (MS WSH).
- **Notify of script blocking** - Vba32 Script-Filter notify a user of every script blocking.

Attention: This option is unavailable in the demo version.

- **Report File**
 - **Keep** - sets the mode when all actions performed by Vba32 Script-Filter as well as their results are written to the report file. Default file is named `Vba32Sck.log` and placed in the Vba32 folder. Press **Browse** to specify another report file name and path.
 - **Detailed** - sets the mode when information about all scanned scripts is written to the report file.
 - **Maximum size, kB** - specifies maximum size of the report file. If this value is achieved, addition of records to the file end will cause deletion of records at the file

beginning. It is recommended to enable it in order not to litter the system with stale data. To view the report file, press **Show**.

6.1.2 Blocking Scripts

If Vba32 Script-Filter detects a malicious script a dialog box notifying about blocking of active script is invoked.

The dialog box contains information about a file (html-page address) that includes the blocked script and the name of virus or another malicious script.

Set **Don't display this message** if you don't want to see the dialog box.

Attention: This option is unavailable in the demo version.

To restore display of the dialog box, set **Notify of script blocking** in the [Vba32 Script-Filter Settings](#) dialog box.

6.2 Vba32 Antidialer

Vba32 Antidialer is a Vba32 component which provides protection against unauthorized attempts to create a connection to unknown phone numbers.


Vba32 Antidialer blocks unauthorized attempts to create a connection to a "strange" provider. Such connections can be created by so called porn-dialers as well as trojans, or other malicious programs. These programs may cause great expenses on using telephone communications.

Note: Since 3.10.5 version Vba32 Antidialer is included in the standard installation package of Vba32 Workstation and Vba32 Personal (see [Vba32 Products](#) for details).

6.2.1 Configuring Vba32 Antidialer

Vba32 Antidialer settings allow specifying actions to perform if an application is attempting to create a connection, forming lists of allowed and forbidden telephone numbers for dialing, configuring the report keeping.

To invoke the dialog box with the **Vba32 Antidialer** settings:

- Right-click the  [tray icon](#) to invoke the [system tray menu](#).
- Select **Settings**.
- Then select **Antidialer**.

Vba32 Antidialer dialog box contains the following tabs:

[General](#)

[Phone numbers](#)

6.2.1.1 General

General tab contains the setting of actions to perform by Vba32 Antidialer when detecting attempts to create a connection and the setting of the report file keeping.

- **Turn on Antidialer** - enables and disables detection of attempts to establish modem connections.
- **Operation mode** - allows selecting one of the dial block modes.

Attention: The operation mode can be changed in registered version only.

- **Don't block numbers** - Vba32 Antidialer doesn't block attempts to create a connection.
- **Block all numbers excluding allowed ones** - Vba32 Antidialer blocks attempts to create a connection by dialing unknown or forbidden phone number.
- **Notify of blocked dial-out attempts** - Vba32 Antidialer notifies on blocking unknown phone numbers by a warning window.

- **Report file**

- **Keep** – sets the mode when all actions performed by Vba32 Antidialer as well as their results are written to the report file. To view the existing report file, press **Show**. To modify current settings of keeping the report file, press **Settings....** The following options can be changed:
- **Save to** - specifies a path where the report file is saved. It's proposed to use the file Vba32ADL.log in the Vba32 folder by default. To specify your own report file, press **Browse...** and type file path and name in the invoked dialog box.
- **Detailed** - sets the mode of logging all dial-up attempts and actions performed by Vba32 Antidialer.
- **Maximum size, Kb** - specifies maximum size of the report file. If this value is achieved, addition of records to the file end will cause deletion of records at the file beginning.

Press **Ok** to save any changes you have made and close the window.
Press **Cancel** to close the window without saving.
Press **Apply** to save any changes you have made without closing the window.

6.2.1.2 Phone numbers

Phone numbers tab contains lists of allowed telephone numbers.

- **List of allowed numbers** - contains numbers or masks which are allowed for dialing.

Attention: When you have created a new connection at your computer, don't forget to add it to the list.

To manage the list, use **Add**, **Remove** and **Modify** buttons.

- To add a new phone number to the list, press **Add**. Specify a phone number and a brief comment regarding it in the invoked dialog box. It's possible to use masks to specify a range of forbidden telephone numbers. For example, if you add **3***, you will allow dialing numbers starting with 3.
- To remove a number from the list, select it and press the **Remove** button.
- To edit a phone number, select it and press **Modify**.

Press **Ok** to save any changes you have made and close the window.
Press **Cancel** to close the window without saving.
Press **Apply** to save any changes you have made without closing the window.

7. Antivirus Quarantine

Functioning of many antivirus programs is based on detection of malicious software by using entries from antivirus bases. Hundreds of new viruses and their modifications are coming into being every day. So even if antivirus bases are updated periodically, there is still a probability to be infected by new malicious programs and computer viruses.

What should be done if an antivirus program has detected a file as a virus, but can't cure it for some reasons, or according to a heuristic analyzer a file is potentially dangerous? If the file contains important data or you have some doubt in its safety, than the best solution is to save a copy of the file to an antivirus quarantine when scanning it.

Antivirus quarantine is a special storage for infected or suspicious files. The files are stored in a special format and can't be executed in the ordinary way that guarantees safety of your computer.

See also:

[Vba32 Quarantine Overview](#)

[Vba32 Quarantine Main Window](#)

[Configuring Vba32 Quarantine](#)

[Working with Quarantined Files](#)

7.1 Vba32 Quarantine Overview

Vba32 Quarantine is a Vba32 component that stores suspicious and infected files placed there by Vba32 components.

When Vba32 detects suspicious or infected file it performs actions according to the specified settings. Provided the **Save copy** option is enabled (Vba32 Monitor, Scanner, Mail Filter and Outlook plug-in), the program will not only perform some actions, but place a copy of the file to Vba32 Quarantine as well. Also, an object can be added to Vba32 Quarantine manually.

Vba32 Quarantine creates a special folder at your computer to isolate suspicious and infected files from the system. Path to the folder can be configured in the [Vba32 Quarantine settings](#) dialog box.


Vba32 Quarantine allows:

- Adding any file on your computer to the storage.
- Doing scanning of files.
- Deleting files from the storage.
- Sending files to the Vba32 antivirus server for detailed analysis.
- Extracting files to the selected folder.
- Restoring files to its initial location.

Note: Since 3.10.5 version Vba32 Quarantine is included in the standard installation packages of Vba32 Workstation, Vba32 Server and Vba32 Personal (see [Vba32 Products](#) for details).

7.2 Vba32 Quarantine Main Window

To invoke the main window of Vba32 Quarantine:

- Right-click the  [tray icon](#) to open [system tray menu](#) .
- Select **Quarantine**.

The main window of Vba32 Quarantine has two tabs: File quarantine and Mail quarantine. File quarantine contains files added by a user, [Vba32 Scanner](#) or [Vba32 Monitor](#). Mail quarantine receives files from [Vba32 Mail Filter](#) and [Vba32 Outlook plug-in](#).

Both File and Mail quarantines display information about files as a table:

- **File name** - full name of a file.

- **Size, bytes** - original size of a file.
- **Information** - virus names of infected files and other extra information.
- **State** – file state. It can be **Clean**, **Suspicious**, **Infected** or **Unknown** (provided a user added it without scanning).
- **Placed** – date and time when a file was placed to Vba32 Quarantine.
- **Sent** - whether a file was sent to the Vba antivirus server for detailed analysis or not (see [Sending Files for Detailed Analysis](#) for details).

Various actions can be performed on quarantined file. For detailed information see [Working With Quarantined Files](#).

7.3 Configuring Vba32 Quarantine

Vba32 Quarantine settings allow specifying a folder to store quarantined files in, a Vba server to upload files for analysis and proxy server configuration. Also, options of automatic maintenance can be configured.

To invoke a dialog box with the **Vba32 Quarantine** settings:

- Invoke the [Vba32 Quarantine Main Window](#)
- Invoke the **Edit** menu.
- Select **Settings....**

The dialog box contains the following tabs:

- [General](#)
- [Maintenance](#)

7.3.1 General

General tab of the [Vba32 Quarantine Settings](#) dialog box contains the following settings:

- **Local storage** - a folder at your computer to store quarantined files in.
- **Remote storage** - a web-address of a Vba server files for detailed analysis are uploaded to.
- **Use proxy-server** - enables using a proxy-server to access **Remote storage**.
 - **Server** - proxy-server address.
 - **Port** - proxy-server port.
 - **User name** - user name to access a proxy-server.
 - **Password** - user password to access a proxy-server.


Press **Ok** to save any changes you have made and close the window.
Press **Cancel** to close the window without saving.
Press **Apply** to save any changes you have made without closing the window.

7.3.2 Maintenance

Maintenance tab of the [Vba32 Quarantine Settings](#) contains settings of storing and maintaining quarantined files:

- **Maintenance period, hours** - sets automatic maintenance of quarantined files in the period of time specified.
- **Maximum Quarantine size, Mb** - enables limitation of the disk space for storage of Vba32 Quarantine files. When the maximum size is exceeded, the older files will be removed until the size decreases to the number specified.
- **Maximum storage time, days** - specifies the maximum time for storing files in Vba32 Quarantine. The files stored longer than the period specified will be deleted from the storage automatically.

Attention: This option is ignored if the program deletes files that exceed the maximum size of the storage.

- **Automatically send suspicious objects** - enables the program to automatically send suspicious files to the VirusBlokAda server for detailed analysis. Its address is displayed on the [General](#) tab.
- **Interactive maintenance** - if enabled, the  icon appears in the system tray menu when maintenance is performed. The **Vba32 Quarantine Maintenance** dialog box is invoked by double-clicking the icon. It displays the current status of maintenance process.

Press **Ok** to save any changes you have made and close the window.
Press **Cancel** to close the window without saving.
Press **Apply** to save any changes you have made without closing the window.

7.4 Working with Quarantined Files

Various actions can be performed on files stored in Vba32 Quarantine. For this, select a file in the table and choose the corresponding action either using the toolbar or the context menu (right-click the files). Action will be applied to all selected files.

The following actions are available:

- **Add** - add any file at your computer to Vba32 Quarantine (see [Adding Files to Vba32 Quarantine](#)).
- **Scan** - scan the files (see [Scanning Quarantined Files](#)).
- **Delete** - remove the files from Vba32 Quarantine (see [Removing Files from Quarantine](#)).
- **Send** - send the files to Vba antivirus server for detailed analysis (see [Sending Quarantined Files for Detailed Analysis](#)).
- **Extract** - save the files to the specified folder (see [Extracting Quarantined Files](#)).
- **Restore** - restore the files to its initial location (see [Restoring Quarantined Files](#)).

7.4.1 Adding Files to Vba32 Quarantine

To add files to Vb32 Quarantine:

Invoke the [Vba32 Quarantine](#) main window.

Select **File**.

Then select **Add...**

Choose files to add and press **Open**.

Wait until the process is completed.

7.4.2 Scanning Quarantined Files

Quarantined files can be scanned or re-scanned. For example, false positives could be fixed after Vba32 update, so some files placed to Vba32 Quarantine as suspicious can be safely restored (see [Restoring Quarantined Files](#)).

To scan one or several quarantined files:

Invoke the [Vba32 Quarantine](#) main window.

Select files in the table (files can be sorted by attributes - file name, size, state, etc).

Select **Scan...** in the context menu of the main window or press the **Scan** button on the toolbar.

The **Scan** dialog box displays the files selected to be scanned. Press the **Settings** button to invoke the [Scanning settings](#) dialog box. Press **Scan** to start scanning.

When scanning is done, press **Close** to close the dialog box.

7.4.2.1 Scanning Settings

The following scanning settings can be configured:

- **Heuristic analysis** - allows detecting unknown malicious programs and modifications of known malicious programs. It provides a more reliable protection of your computer. There are four levels of heuristic analysis:
 - **Disabled** - unknown malicious programs will not be detected.
 - **Optimal** - practically doesn't slow down the scanning. Recommended for most of users.
 - **Maximum** - provides the maximum level of unknown malicious programs detection with the lowest probability of false positives; slows down scanning a bit.
 - **Excessive** - detects most of unknown malicious programs with the highest probability of false positives. Recommended for advanced users only.

Attention: Send suspicious files to newvirus@anti-virus.by for detailed analysis. This will help us in removing false positives in the next update of antivirus base.

- **Scan mail** - mail bases and messages are scanned.
- **Detect installers of malware** - installers of malicious programs and computer viruses are detected.
- **Scan archives** - archived files are scanned.
- **Thorough mode** - sets the excessive mode of file scanning.

Attention: Thorough mode may considerably increase the time of file scanning.

- **Detect Spyware, Adware, Riskware** - applications of Adware and Riskware types are detected. They are considered as common infected files.
- **Change object status during scanning** - allows changing the **Information** column of the table in the [Vba32 Quarantine](#) window.

Press **Ok** to save any changes you have made and close the window.
 Press **Cancel** to close the window without saving.
 Press **Apply** to save any changes you have made without closing the window.

7.4.3 Removing Files from Vba32 Quarantine

Files can be removed from Vba32 Quarantine if they aren't needed any more. To perform this: Invoke the [Vba32 Quarantine](#) main window.

Select files in the table (files can be sorted by attributes - file name, size, state, etc).

Select **Remove...** in the context menu of the main window or press the **Remove** button on the toolbar.

Press **Yes** to confirm the removal.

Attention: Removed files won't able to be restored. Be attentive while removing files!

7.4.4 Sending Quarantined Files for Detailed Analysis

Selected files can be sent to a special Vba server for detailed analysis. Use this feature in the following cases:

- An infected file was placed to Vba32 Quarantine by Vba32 Scanner or Vba32 Monitor because it had failed to be cured.
- An unknown file recieved by e-mail and placed to Vba32 Quarantine as suspicious.
- A file is a false positive for sure.

Don't forget to specify your valid e-mail address for feedback. Specify as many details as possible in the **Note** text field: version of your operating system, installed service packs, Vba32 package version, date of the last update and reasons of sending files. Specialists of VirusBlokAda will explore your files and send you the result of analysis.

To send one or several quarantined files for detailed analysis:

Invoke the [Vba32 Quarantine](#) main window.

Select files in the table (files can be sorted by attributes - file name, size, state, etc).

Select **Send...** in the context menu of the main window or press the **Send** button on the toolbar.

The **Send files** dialog box contains files to be sent.

- Specify your e-mail address for feedback
- Give detailed description of the files you are sending.
- Press the **Send** button to send the files to the server.

Press **Close** to cancel sending and close the dialog box.

7.4.5 Extracting Quarantined Files

To extract files:

Invoke the [Vba32 Quarantine](#) main window.

Select files in the table (files can be sorted by attributes - file name, size, state, etc).

Select **Extract to...** in the context menu of the main window or press the **Extract** button on the toolbar.

Select a folder to extract the file to or create a new one in the **Browse for Folder** dialog box.

Press **OK**.

Attention: Quarantined files may contain malicious programs. Be careful while extracting!

7.4.6 Restoring Quarantined Files

Restoring allows placing quarantined files to its initial location in your system.

To restore files:

Open the [Vba32 Quarantine](#) main window.

Select files in the table (files can be sorted by attributes - file name, size, state, etc.).

Select **Restore...** in the context menu of the main window or press the **Restore** button on the toolbar.

Press **Yes** to confirm restoring.

To terminate restoring, press **Terminate** button.

Attention: Quarantined files may contain malicious programs. Be careful while restoring!

8. SendLogs Utility

The utility is supplied with every [Vba32 product](#). It is designed to collect technical information and report files of Vba32 modules and either send them to **VirusBlokAda** specialists or save on a disk.

The information simplifies the support and allows the developers to respond promptly to user requests.

See also:

[Starting SendLogs](#)

[Welcome Screen](#)

[Collecting Report Files](#)

[Methods of Sending](#)

8.1 Starting SendLogs

SendLogs is supplied with every [Vba32 product](#).

It can be launched by one of the following ways:

- Switch to a folder where Vba32 is installed to and run the SendLogs.exe file.

Or

- Do as follows:
 - Invoke the [Vba32 Loader main window](#).
 - Click the **Support** link on the [General](#) tab.
 - Press the **Request support** in the invoked dialog box to launch **SendLogs**.

8.2 Welcome Screen

The SendLogs welcome screen contains brief description of the utility.

Press **Next** to proceed to the next step.
Press **Cancel** to cancel sending of logs.

8.3 Collecting Report Files

The **Collecting report files** dialog box contains a progress bar, that displays the status of file collecting, the **Show files** button, that invokes a dialog box with the list of files prepared to be sent. Also, after all files has been collected, the dialog box displays the size of entire attachment.

Press **Back** to return to the previous step.
Press **Next** to proceed to the next step.
Press **Cancel** to cancel sending of logs.

8.4 Methods of Sending

The **Selecting method to send logs** dialog box allows selecting one of three methods of sending:

- [Send logs directly to technical support server](#)
- [Send logs by installed mail client](#)
- [Save logs on a disk](#)

8.4.1 Send Logs Directly

Files will be sent to support server directly from SendLogs by SMTP.

Provided this method is selected, the **Specifying data** dialog box is invoked and it contains text fields to specify e-mail, where the reply for user request will be sent to, subject and body of the request. Also, the dialog box supports the `drag and drop` technology, so it's possible to add files, which will be attached to the e-mail, by dragging them to the dialog box (a default attachment is `Vba32Logs.zip`).

Press **Back** to return to the previous step.
Press **Cancel** to cancel sending of logs.

8.4.2 Send Logs by Mail Client

Provided this method is selected, the utility creates a mail message and invokes installed mail client to send it.

8.4.3 Saving Logs

Provided this method is selected, the utility creates an archive with files to be sent and saves it to user temporary folder. Then the Windows Explorer window, which displays the `Vba32Logs.zip` archive, is invoked.

9. Updating Vba32

Variety of malicious programs is coming into being every day. It may cause single cases of infection as well as mass epidemics with millions computers infected. So regular antivirus updates are crucial nowadays.

VirusBlokAda Ltd traces carefully the creation of new viruses and releases updates of antivirus bases and Vba32 modules on-the-fly.

See also:

[Update](#)

[Viewing Update Results](#)

9.1 Update

To provide reliable protection of your computer against new malicious programs, it's recommended to update Vba32 regularly. Vba32 updates are needed to get the latest versions of Vba32 antivirus bases and components.

The following update issues are described:

[Update Files](#)

[Automatic Update](#)

[Manual Update](#)

9.1.1 Update Files

Updates of antivirus bases are the files that expand the list of known malicious programs and computer viruses. Vba32 can't detect and cure new viruses efficiently without them.

New versions of the Vba32 components are more advanced and improved than the old ones. Files of Vba32 components updates are called binary patches.

A binary patch contains only that part of a component which has been updated. This helps in reducing amount of data to download. After a binary patch has been downloaded the update program adds changes from it to a component being updated, thus the old version is substituted with the new one.

9.1.2 Automatic Update


Automatic update is recommended for most of users. It allows downloading [updates of Vba32 antivirus bases](#) and [updates of Vba32 modules](#).

To change automatic update settings:

- Invoke the [Vba32 Loader main window](#).
- Switch to the [Additional](#) tab.
- Specify the following parameters of the automatic update:
 - **Time intervals, hrs** - enables automatic Vba32 update. The period of time between updates (3 hours by default) can be changed.

Attention: Do not disable automatic update with no reason and for a long time. Only regular updates of Vba32 provide reliable protection of your computer against malicious programs and computer viruses.

- **Interactive** - enables the interactive mode of automatic update. The update process is displayed in a dialog box and every action is submitted with a user.
- Press **Apply** to save any changes you have made.

Automatic update is performed as soon as specified time interval is up (time of the next automatic update is displayed in **Next** text field). The  icon in the [system tray menu](#) indicates that automatic update is running.

9.1.3 Manual Update

Manual update can be started at any time with or without automatic update enabled.

To start update manually:

- Invoke the [Vba32 Loader main window](#).
- Switch to the [Additional](#) tab.
- Press the **Update** button.

or

- Invoke the [system tray menu](#).
- Select **Update**.

9.2 Viewing Update Results

See also:

[Update Window](#)

[Report File](#)

9.2.1 Update Window

Update window is invoked when updating manually or when updating automatically, provided the **Interactive** option is set. Detailed information about update is written to a [report file](#).

9.2.2 Report File

All actions regarding the update process are written to a report file.

To view the report file:

- Invoke the [Vba32 Loader main window](#).
- Switch to the [Additional](#) tab.
- Press the **Show** button in the Report File section.

10. Vba32 Security

Vba32 security is provided by using password protection of settings and actions.

See also:

[Using Password](#)

[Setting Password](#)

[Changing Password](#)

[Removing Password](#)

10.1 Using Password

Password usage prevents Vba32 from being compromised by intruders. Antivirus protection administrator is ensured that users can't manage Vba32 on their own, modify its settings and disable antivirus protection.

Use the following recommendations regarding passwords:

It's recommended using passwords which include at least six characters.

When specifying a password don't use:

- months, years, days of the week, etc.;
- surnames, initials, car registration numbers;
- names of companies and their identifiers;
- telephone numbers or groups of numbers;
- more than two similar numbers following one after another;
- a string of characters that contains numbers or letters only;

It's recommended changing passwords regularly (e.g. once a month) and not using old passwords.

Keep passwords secret.

Change passwords whenever you think they have been compromised.

10.2 Setting Password

To set a password:

Invoke the [Vba32 Loader main window](#).

Switch to the **General** tab.

Press the **Password** button.

Leave the **Old password** text field empty.

Type a new password in the **New password** text field.

Retype the password in the **Confirm new password** text field.

Press **Ok** to save changes you have made and close the dialog box.

10.3 Changing Password

To change a password:

Invoke [Vba32 Loader main window](#).

Switch to the **General** tab.

Press the **Password** button.

Type old password in the **Old password** text field.

Type a new password in the **New password** text field.

Retype the password in the **Confirm new password** text field.

Press **Ok** to save changes you have made and close the dialog box.

10.4 Removing Password

To remove a password:

Invoke the [Vba32 Loader main window](#).

Switch to the **General** tab.

Press the **Password** button.

Type old password in the **Old password** text field.

Leave the **New Password** and **Confirm New Password** text fields empty.

Press **Ok** to save changes you have made and close the dialog box.

11. Activating Vba32

Vba32 Activation allows a user to switch the program from demo mode to the full one. [A special utility](#) collects some user information, sends it to an activation server, waits until a key file is received and installs it in the system.

11.1 Launching Vba32 Activation

There are several ways of launching the activation:

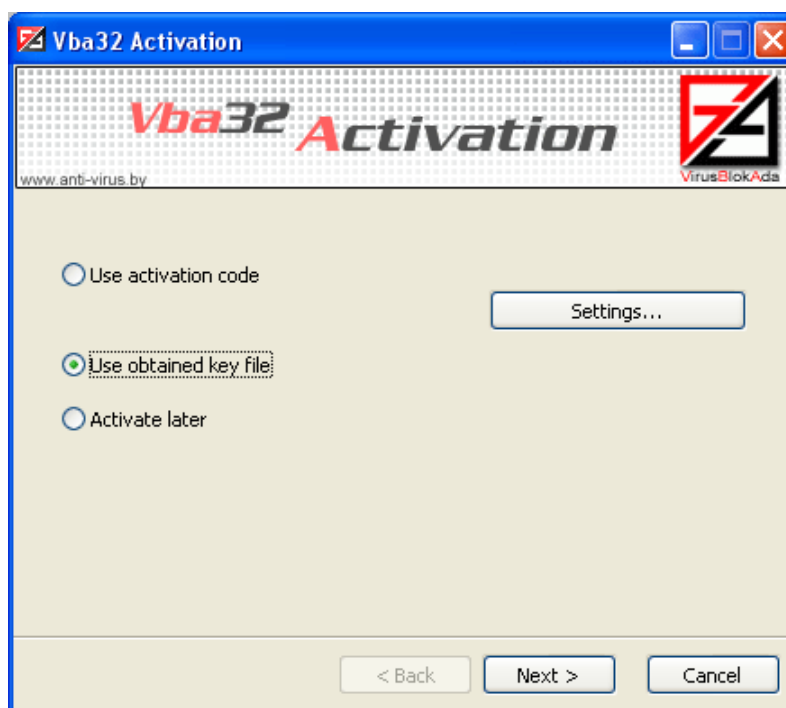
- Select the corresponding item in the [system tray menu](#).
- Press the **Activate** button on the warning dialog box which reminds of forthcoming key expiration.
- Press the **Activate** button on the warning dialog box which notifies that a key file is missing or expired.

A user can choose from two ways of activation:

- [Using key file](#).
- [Using activation code](#).

11.2 Using Key File

Users can activate Vba32 by a key file that has been obtained before. For this, the **Use obtained key file** menu item on the utility main window should be selected.

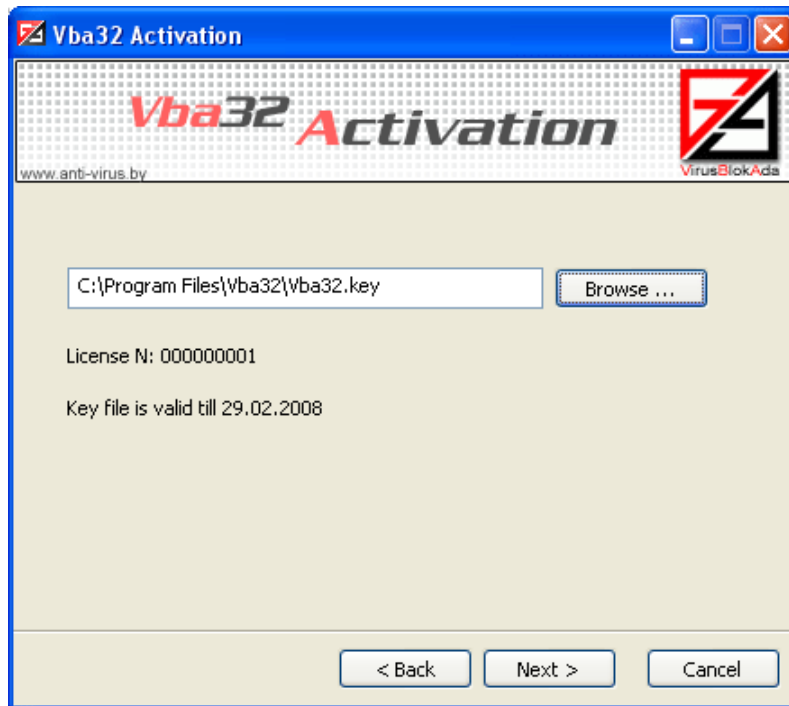


Choosing Vba32 activation method

By pressing the **Next** button a user proceeds to the next step of the activation - [Browsing for key file](#).

11.2.1 Browsing for Key File

User can specify a **Vba32** key file that has been obtained before. Pressing the **Browse...** invokes the standart Windows open file dialog box. After the key file has been selected, information on it (license number and its period of validity) will be displayed on the dialog box.



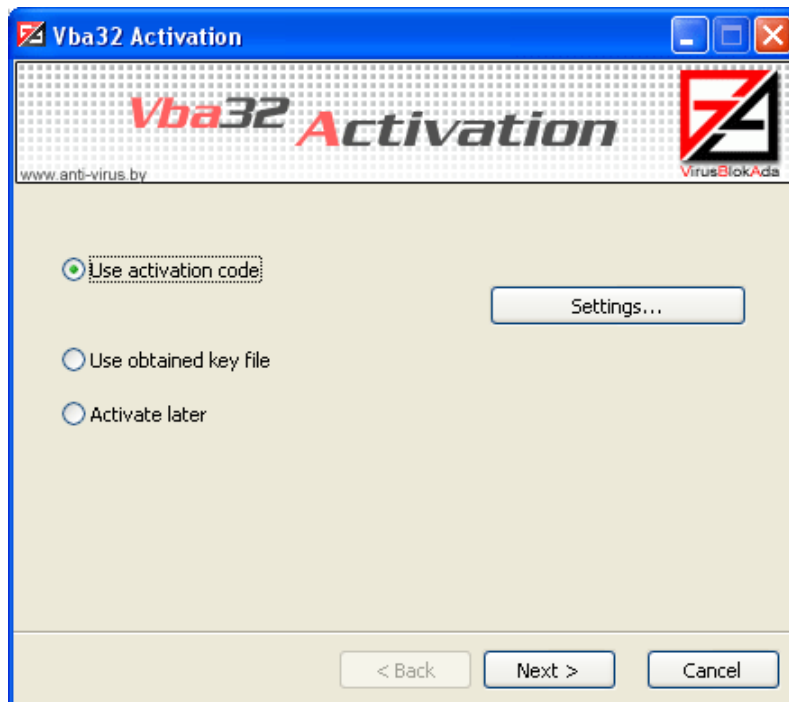
Dialog box of specifying a key file

The **Next** button is disabled unless the path to a key file is specified.

Press **Back** to return to the previous step of the activation.
Press **Next** to proceed the activation.
Press **Cancel** to cancel the activation.

11.3 Using Activation Code

Users can activate Vba32 by a special code, which has been obtained when buying **Vba32**.



Choosing Vba32 activation method

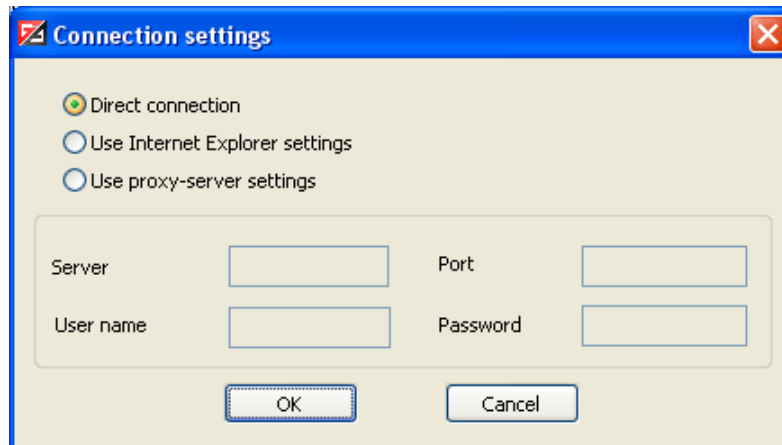
Vba32 activation code contains 4 groups in 5 symbols (digits and latin letters) as xxxxx-xxxxx-xxxxx-xxxxx. It's unique and registered in the database of the activation server.

This activation method consists of the following steps:

- [Configuring connection](#).
- [Specifying user information](#).
- [Receiving key file](#).

11.3.1 Configuring Connection

Press the **Settings...** button on the utility main window, to invoke the **Connection settings** dialog box.



Configuring connection setting

There are three ways of connecting to an activation server:

- **Direct connection** - direct `HTTP` connection is used to connect to the activation server.
- **Use Internet Explorer settings** - connecting setting specified in Internet Explorer is used to connect to the activation server.
- **Use proxy - server settings** - enables using proxy - server to connect to the activation server:
 - **Server** - proxy - server address.
 - **Port** - proxy - server port.
 - **User name** - user name that will be used when connecting to the proxy - server.
 - **Password** - user password that will be used when connecting to the proxy - server.

Press **OK** to apply the setting and close the dialog box.
Press **Cancel** to close the dialog box without applying the settings.

11.3.2 Specifying User Information

After choosing **Using activation code** on the utility main window and [configuring connection settings](#), user should specify the following information:

- **Activation code** - edit box where user should specify the activation code obtained when buying **Vba32** or by any other way.
- **User** - edit box where user should specify his name.
- **E-mail** - edit box where user can specify a e-mail address that will be used to send all useful information regarding Vba32.
- **Phone number** - phone number that should be used by Vba32 support team to contact user.
- **Country** - list where user can select a country. If it doesn't contain the needed one, than it's recommended to specify the **Other** value.

The **Send data to activation server** flag allows sending all information to the activation server.

Vba32 Activation

www.anti-virus.by

VirusBlokAda

Activation code: XXXXX-XXXXX-XXXXX-XXXXX

User: Dmitry

E-mail: support-en@anti-virus.by

Phone number: +375172266285

Country: Other

Send data to activation server

< Back Next > Cancel

Specifying user information

[VirusBlokAda Ltd.](#) strongly recommends you registering. A registered user will get additional technical support, notifications and useful information. Moreover after registering users receive e-mail containing contacts of a company which they can request for support.

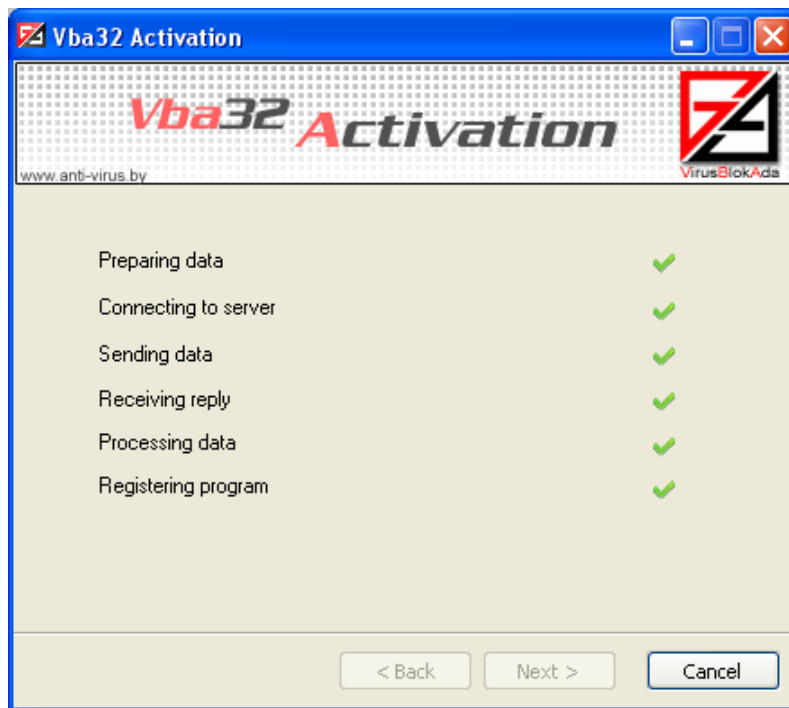
Press **Back** to return to the previous step of the activation.
Press **Next** to proceed the activation.
Press **Cancel** to cancel the activation.

11.3.3 Receiving Key File

After user information [has been specified](#), the utility receives a key file and installs it in the system. The status of the following operations is displayed during this:

- Preparing data
- Connecting to server
- Sending data
- Receiving reply
- Processing data
- Registering program

Provided all labels are green, operations have been successfully finished. If at least one is red, the activation has been failed, at that a warning dialog box, that contains error description, appears. In this case it's recommended performing the activation again or contact [technical support team](#) (the situation should be described in details in the request).

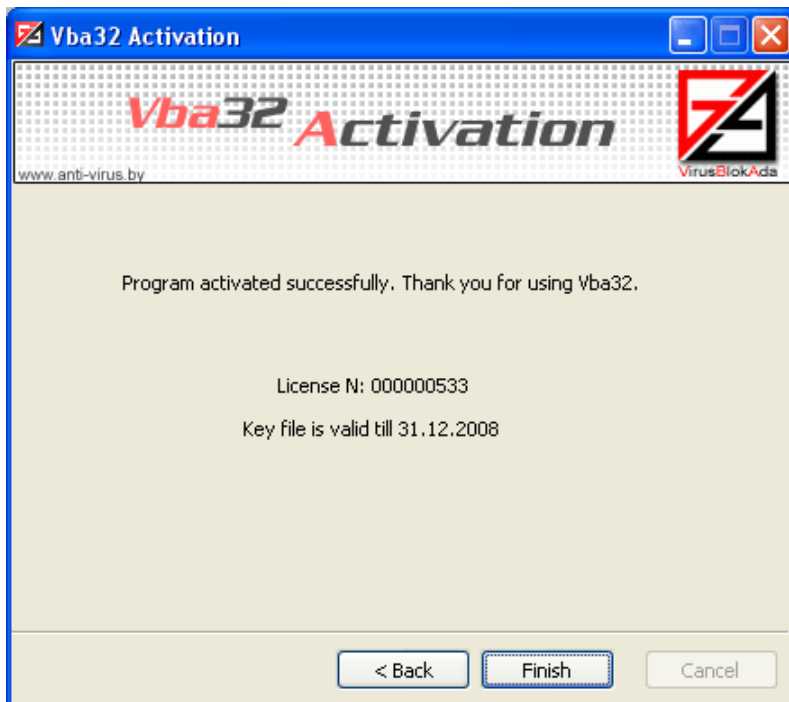


Communication of Vba32 activation utility with the activation server

Press **Back** to return to the previous step of the activation.
Press **Next** to proceed the activation.
Press **Cancel** to cancel the activation.

11.4 Finishing Activation

When all operations on the activation have been successfully performed the following window appears. It also contains information on license number and key file period of validity.



Vba32 activation has been successfully finished

Press **Finish** to finish the activation

12. Vba32 Scheduler

Vba32 Scheduler is developed to launch the programs (applications, scanning, update) in the specified time periodicity on personal computers, workstations and servers.

The following items explain the right work with task scheduling:

- **Create**
- **Edit**
- **Delete**
- **Run**

Additional facilities of Vba32 Scheduler:

- **View logs**
- **Detailed view**

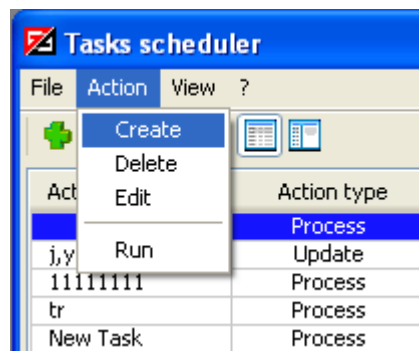
Settings main facilities:

- [Action types of tasks](#)
- [Scheduling of time periodicity](#)


12.1 Main and additional facilities of Vba32 Scheduler

12.1.1 Create task

To create task it's necessary to do the following:



Launch the task from menu

- Select action **Create** in menu or click  **Create**.
- Enter the name of the task, select [Action type](#) and press **Next-->**

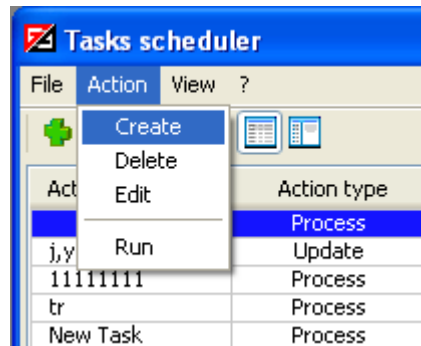
The name of the task can not be repeated with already existing tasks.

- If necessary, configure the action task and click **Next-->**
- Select [time periodicity](#) and click **Next-->**
- If necessary, configure periodicity of task launch and click **Finish**.


12.1.2 Edit task

To edit task it's necessary to do the following:

- Select task from the list of actions



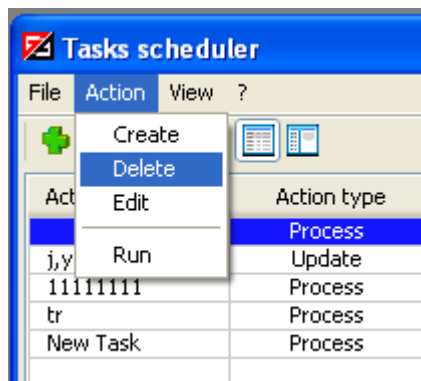
Edit the task from menu

- Select action **Edit** in menu or click  **Edit**.
- В появившемся окне возможно редактирование параметров задачи (task name, [action types](#), [time periodicity](#))
- To apply settings click **Apply** or **OK**, to undo it's necessary to click **Cancel** or close the window.


12.1.3 Delete task

To delete task it's necessary to do the following:

- Select the task from the list.



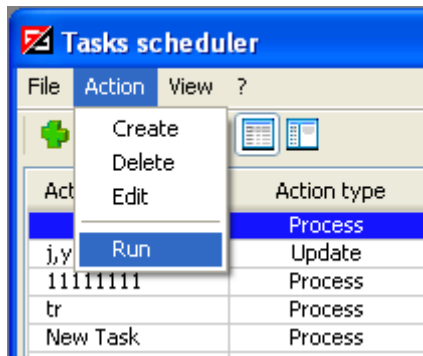
Deletion the task from menu.

- Select action Delete in menu or click  Delete.
- Confirm the action in dialog window.


12.1.4 Run task

To run the task forcibly:

- Select the task from the list.




Run the task from menu

- Select action Run in menu or click  **Run**.

Forced launches doesn't influence on periodicity. The information about them is stored only in logs.

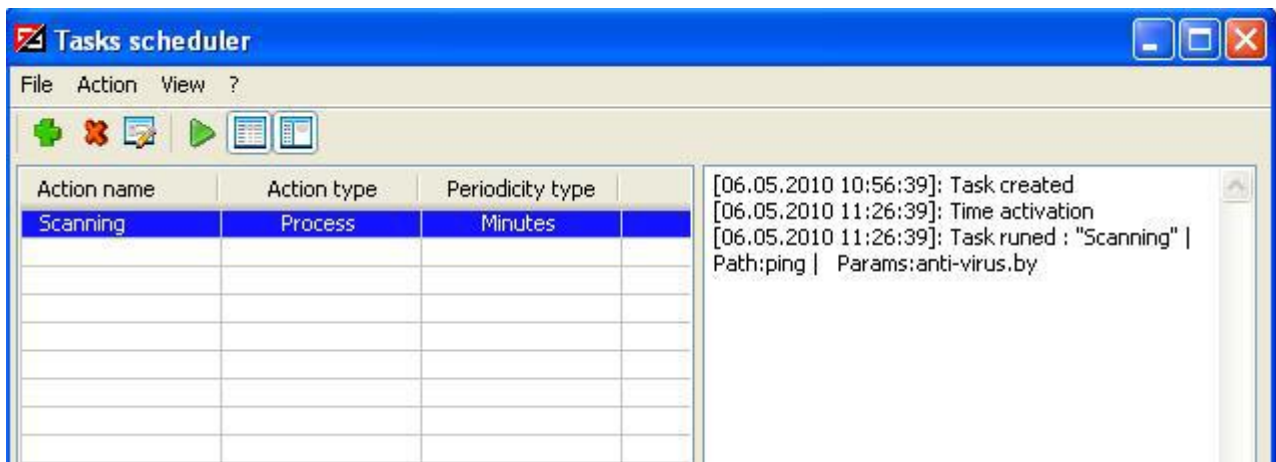
12.1.5 Log View

To view the task log it's necessary to do the following:

Select Log View in menu View or click the button  Log View

Select task from the list


In the right part of the window it will be displayed log of selected task



Log View

12.1.6 Detailed task mapping

To detail task mapping it's necessary to do the following:

Select Detailed Mapping in the menu or click  **Detailed Mapping**

In the task list it will be displayed the additional information

Action name	Action type	Periodicity type	Path	Parameters	Last launch	Next launch	Enable/Disable
j,yjdk	Process	Fixed date	am Files\yba32\yba3	/mn	7\07\2010 17:03:4		<input checked="" type="checkbox"/>
11111111	Update	Minutes	am Files\yba32\yba3	/BL	7\07\2010 12:13:2	7\07\2010 12:18:2	<input checked="" type="checkbox"/>
tr	Process	Fixed date	ping	anti-virus.by	1\07\2010 11:27:4		<input checked="" type="checkbox"/>
New Task	Process	Days	:\Spell Checker 2\Spe	anti-virus.by	6\07\2010 08:21:0	6\07\2010 15:30:3	<input checked="" type="checkbox"/>
	Process	Days	:\Spell Checker 2\Spe	anti-virus.by	7\07\2010 08:47:5	28\07\2010	<input checked="" type="checkbox"/>

Detailed task mapping.

12.2 Action types

Task name

Process
 Scan
 Update

Select action type

Vba32 Scheduler can create tasks with the following action types:

Process is launching of user process.

Scan is launching of console scanner.

Update is launching of update process.

12.2.1 Process

Process is a process that is being configured by user.

Path

Parameters

Process settings

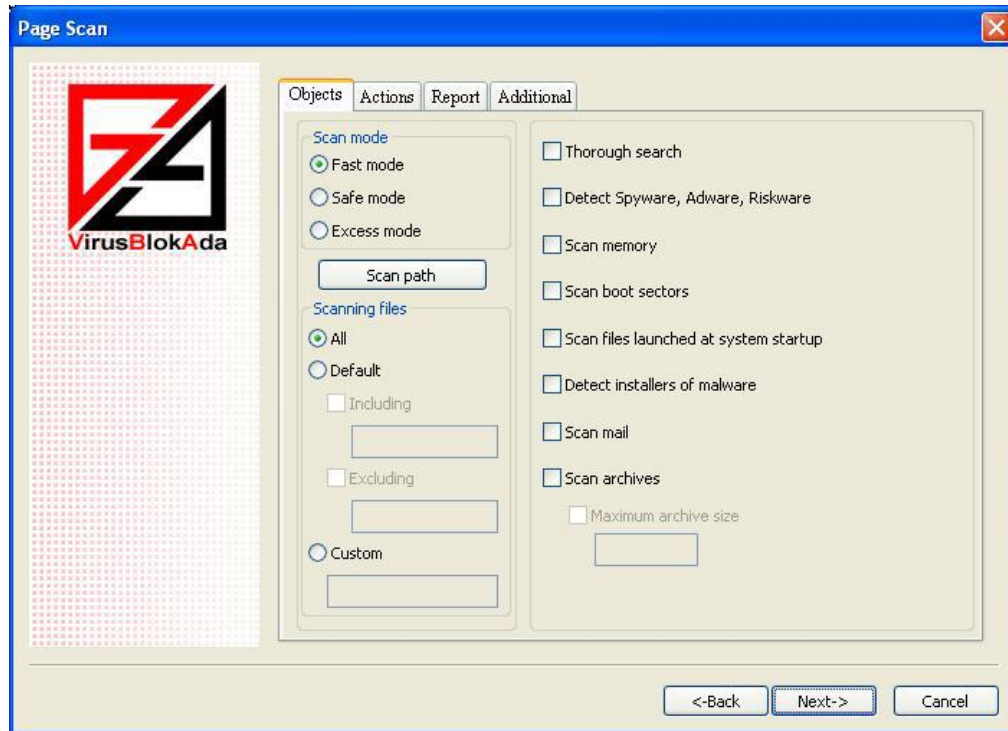
Path is a path to user process.

Parameters is parameters that are passed via command line when the process launches.

12.2.2 Scanning

Scanning – launching of console scanner.

Settings:

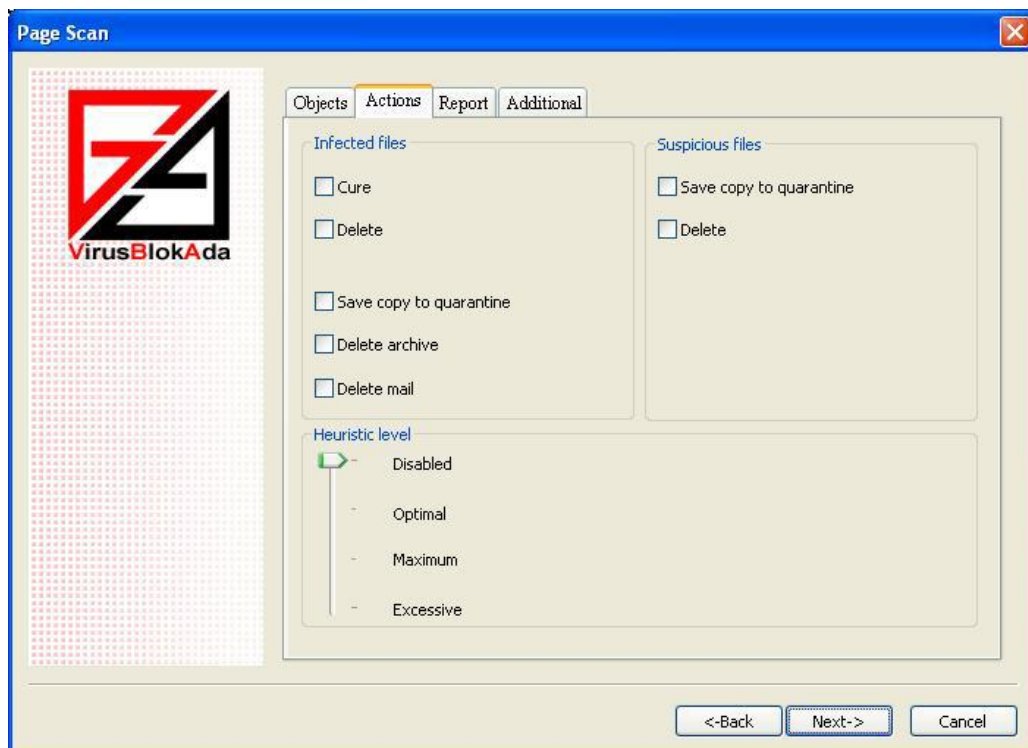


Objects scanning settings

- **Object** - objects scanning settings include :
 - **Scan mode:**
 - **Fast** (flag /M=1)
 - **Safe** (flag /M=2)
 - **Excessive** (flag /M=3)
 - **Scan path** is a dialog box with the ability to add, edit and delete scanning paths.
 - **Scan types:**
 - **All**
 - **Default**
 - **Including** (flag /EXT+''')
 - **Excluding** (flag /EXT-''')
 - **Custom** (flag /EXT=''')
 - **Thorough search** (флаг /PM)
 - **Detect Adware, Spyware, Riskware** (flag /RW)
 - **Scan memory** (flag /MR)
 - **Scan boot sectors** (flag /BT)
 - **Scan files launched as system startup** (flag /AS)
 - **Detect installers of malware** (flag /SFX)

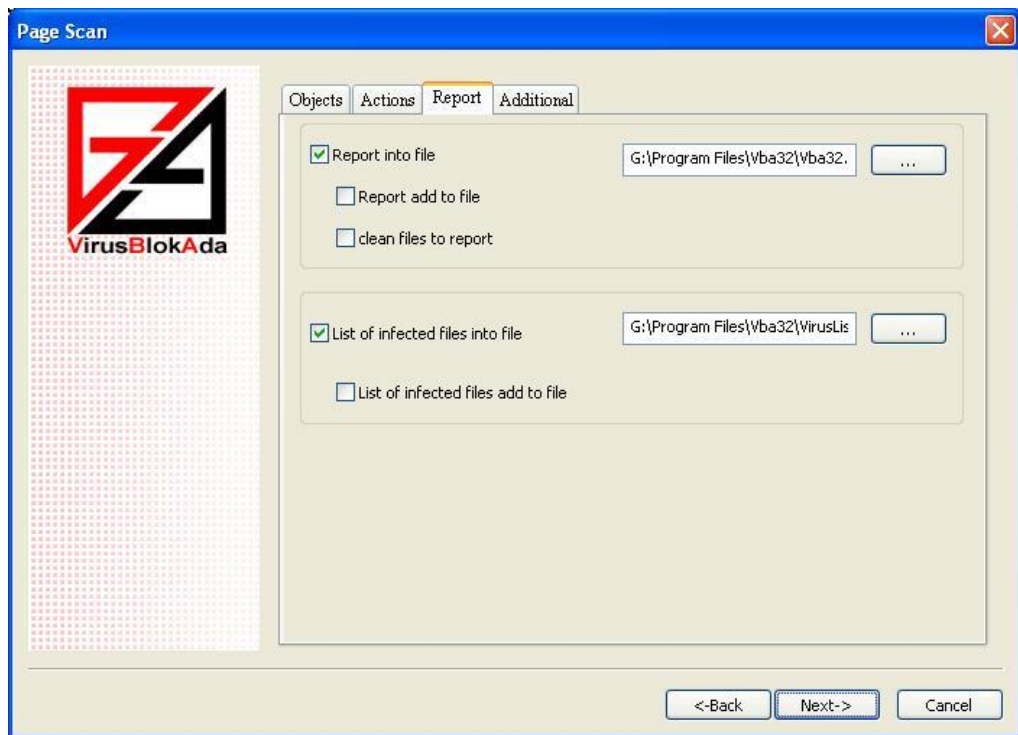
- **Scan mail** (flag /ML)
- **Scan archives** (flag /AR)

Maximum archive size (flag /AL=)



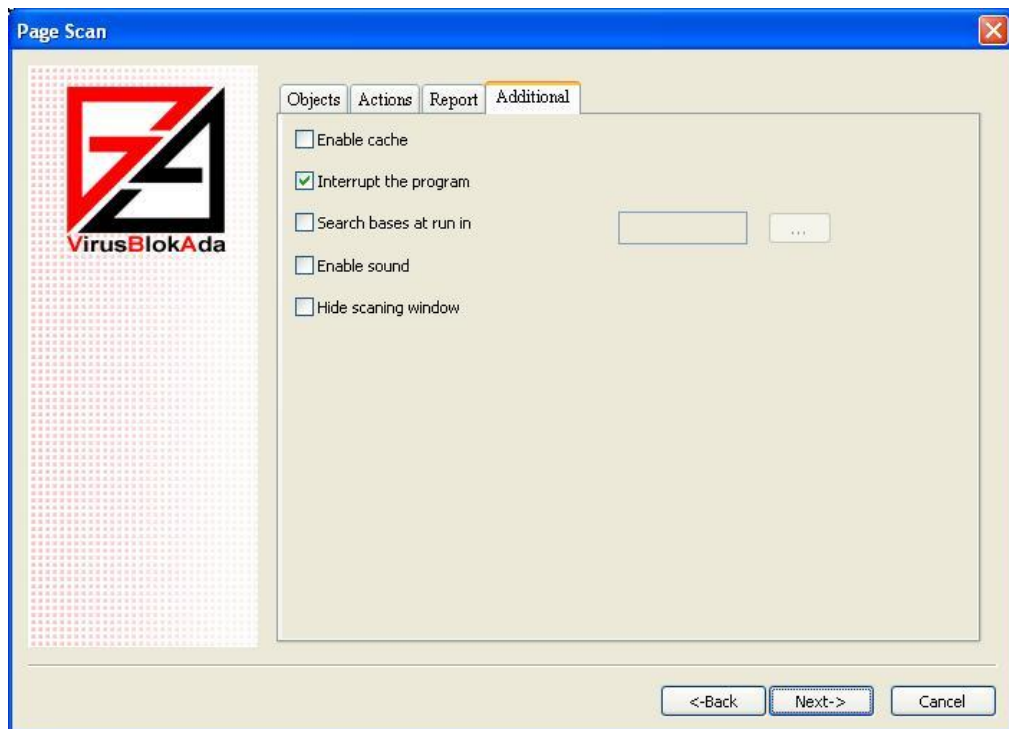
Actions on objects settings

- **Actions on objects include**
 - **Infected files**
 - **Cure** (flag /FC)
 - **Delete** (flag /FD)
 - **Save copy to quarantine** (flag /QI)
 - **Delete archive** (flag /AD)
 - **Delete mail** (flag /MD)
 - **Suspicious files**
 - **Save copy to Quarantine** (flag /QS)
 - **Delete** (flag /SD)
 - **Heuristic Analyses**
 - **Disabled** (flag /HA=0)
 - **Optimal** (flag /HA=1)
 - **Maximum** (flag /HA=2)
 - **Excessive** (flag /HA=3)



Report scanning settings

- **Report** – report scanning settings
 - **Keep report into file** (flag /R=)
 - **Add report into file** (flag /R+)
 - **Include names of "clean" files into report** (flag /OK)
 - **Keep list of infected files into file** (flag /L=)
 - **Add list of infected files into file** (flag /L+)



Configuring additional parameters of scanning

- **Additional** - configuring additional parameters of scanning
 - **Enable cache** (flag /CH)

- **Disable the program** (flag /QU)
- **Search startup database updates** (flag /DB=)
- **Enable sound warning** (flag /SS)
- **Hide scanning window** – the scanning process will be created without a window.

12.2.3 Update

Update is update of Vba32 antivirus.

Settings of update path and access to the network are located in [Additional](#) tab of Vba32 Loader main window.

12.3 Sheduling of launch time

Scheduler have the ability to launch tasks with the periodicity:

- [Minutes](#) - launching the task in definite number of minutes.
- [Hours](#) - launching the task in definite number of hours.
- [Days](#) - launching the task in definite number of days with the possibility of launching missed task and running the task in definite or random time.
- [Weeks](#) - launching the task in definite days of the week with the possibility of launching missed task and running the task in definite or random time.
- [Months](#) - launching the task in definite days of the month with the possibility to run missing action and to launch it in definite or random time of the day.
- [Fixed date](#) - launching the task in definite date and time with the possibility to run missing action.

12.3.1 Scheduling of launch time: Minutes

Minutes - launching the task in definite number of minutes.

It is possible to insert any number from 1 to 1399.

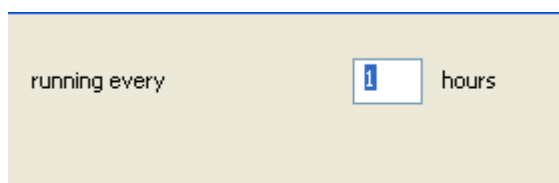


Launching time setting: Minutes

12.3.2 Scheduling of launch time: Hours

Hours - launching the task in definite numbers of hours.

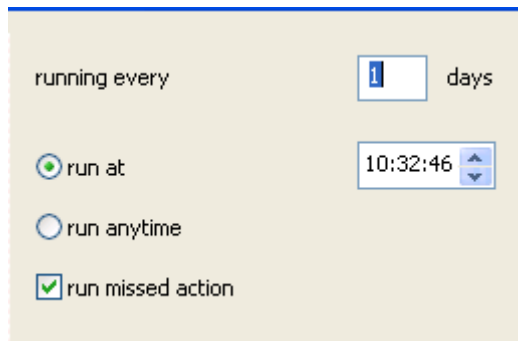
It is possible to insert any number from 1 to 159.



Launching time settings: Hours

12.3.3 Scheduling of launch time: Days

Days - launching the task in definite number of days with the possibility of launching missed task and running the task in definite or random time.



running every days

run at

run anytime

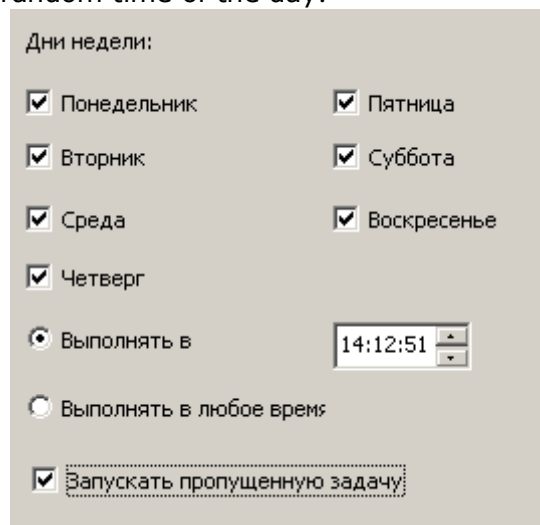
run missed action

Launching time settings: Days

- **Run at...** - the task runs at definite time. It is possible to insert any number from 1 to 61.
- **Run anytime** - the task runs at the first opportunity.
- **Run missed actions** - this item gives the ability to launch the task which for some reason was missed.

12.3.4 Scheduling of launch time: Weeks

Weeks - launching the task in definite days of the week with the possibility to run missing action and to startup in definite or random time of the day.



Дни недели:

Понедельник Пятница

Вторник Суббота

Среда Воскресенье

Четверг

Выполнять в

Выполнять в любое время

Запускать пропущенную задачу

Launching time settings: Weeks

- **Weekdays** - days in which the launch of task is necessary. It should be specified at least one day.
- **Run at...** - the task runs at definite time.
- **Run anytime** - the task runs at the first opportunity.
- **Run missed actions** - this item gives the ability to launch the task which for some reason was missed.

12.3.5 Scheduling of launch time: Month

Days of the months - launching the task in definite days of the month, with the possibility to launch the task which for some reason was missed at definite or random time of the day.

Days of month:

1 14 22 28 30

run at 14:49:46

run anytime

run missed action

Launching time settings: Months

- **Days of the months** - days when it's necessary to launch the task. It's possible to enter any numbers from 1 to 31 separated by spaces and/or by comma.
- **Run at...** - this item gives the ability to select definite time of task launch.
- **Run anytime** - this item let the task launch at random time.
- **Run missed actions** - this item gives the ability to launch the task at the first opportunity which for some reason was missed.

12.3.6 Scheduling of launch time: Fixed date

Fixed date is launching the task in definite date and time with the possibility to run missed action and to run in definite or anytime.

< August 2010 r. >

Пн	Вт	Ср	Чт	Пт	Сб	Вс
26	27	28	29	30	31	1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	31	1	2	3	4	5

14:49:46

run missed action

Launching time settings: Fixed Date

Run missed actions - if it wasn't possibility to launch the task at fixed time it would run at the first opportunity.