AVG 2013 logs & troubleshooting guide

Contents

[1. AVG client log files 3](#_Toc333910957)

[1.1. Location of the logs: 5](#_Toc333910958)

[2. Encoded & Decoded logs 6](#_Toc333910959)

[3. How to decrypt received log files 7](#_Toc333910960)

[4. Structure of the encrypted log records 8](#_Toc333910961)

[4.1. Logged component (Name of the logger) 8](#_Toc333910962)

[4.2. Severity 8](#_Toc333910963)

[INFO 8](#_Toc333910964)

[DEBUG 8](#_Toc333910965)

[DETAIL 9](#_Toc333910966)

[WARN 9](#_Toc333910967)

[ERROR 9](#_Toc333910968)

[4.3. Date and time 9](#_Toc333910969)

[4.4. Message text 9](#_Toc333910970)

[5. How to change the logging level 10](#_Toc333910971)

[6. Numbering and rolling of log files 11](#_Toc333910972)

[7. Translation of error codes 12](#_Toc333910973)

[8. Examples of some logs 13](#_Toc333910974)

[8.1. Personal e-mail scanner 13](#_Toc333910975)

[8.2 Install log 14](#_Toc333910976)

[parameters used during installation - mfa\*.log 14](#_Toc333910977)

[license number used for installation - mfa\*.log 14](#_Toc333910978)

[version of AVG installation package - msi\*.log 14](#_Toc333910979)

[user who launched the installation - msi\*.log 14](#_Toc333910980)

[installation directory - msi\*.log 15](#_Toc333910981)

[8.3 Update log 20](#_Toc333910982)

[9. AVG diagnostic packages 21](#_Toc333910983)

[9.1. AVGDIAGX output 21](#_Toc333910984)

[9.2. FWDIAG output 22](#_Toc333910985)

[10. AVG issues and important log files 23](#_Toc333910986)

[10.1. Update problems 23](#_Toc333910987)

[10.2. Problem with scheduled tasks 23](#_Toc333910988)

[10.3. Problems with Resident Shield 23](#_Toc333910989)

[10.4. Problems with AVG scans 23](#_Toc333910990)

[10.5. Problems with Online Shield (or LinkScanner) 23](#_Toc333910991)

[10.6. Problems with AVG gui (crashes, unexpected moving, closing, etc) 24](#_Toc333910992)

[10.7. Problems with Remote Administration 24](#_Toc333910993)

[10.8. Installation problem 24](#_Toc333910994)

[10.9. Problem with E-mail scanner 24](#_Toc333910995)

[10.10. Problem with Outlook plugin or The Bat! plugins 24](#_Toc333910996)

[10.11. Problems with System Tools component 24](#_Toc333910997)

[10.12. Problems with Identity Protection component 25](#_Toc333910998)

[10.13. Problems with Anti-Spam 25](#_Toc333910999)

[10.14. Firewall issues and important log files 25](#_Toc333911000)

[Firewall is blocking some application 25](#_Toc333911001)

[Firewall failed to start 25](#_Toc333911002)

[Firewall crashed or causes crash 26](#_Toc333911003)

[Firewall is not blocking any specific communication 26](#_Toc333911004)

[11. AVG Admin log files 27](#_Toc333911005)

[11.1. Location of the logs: 27](#_Toc333911006)

[12. AVG Admin issues and important log files 29](#_Toc333911007)

[Creation of DataBase fails 29](#_Toc333911008)

[AVG Admin Servers can not start 29](#_Toc333911009)

[Network installation fails 29](#_Toc333911010)

[Centralized update not working 29](#_Toc333911011)

[Synchronization not working 29](#_Toc333911012)

[13. Configuration 30](#_Toc333911013)

[13.1. General information 30](#_Toc333911014)

[13.2. Location of AVG configuration files 30](#_Toc333911015)

1. AVG client log files

AVG produces following log files:

* **admincli.log** – related to the remote administration component
* **advisor.log** – related to AVG advisor
* **avgapi.log** – related to API. Some of our partners are using api calls to AVG to control various settings (e.g. disable RS, verify component state). These api calls and related information are written to this log. Log is created only if avgapi.cfg exists. CFG is not created by setup, it must be created manually.
* **avgasfl.log** - Antispam on Exchange 2003 private
* **avgasflp.log** - Antispam on Exchange 2003 public
* **avgavtar.log** - Routing transport agent private log
* **avgavtarp.log** - Routing transport agent public log
* **avgavtas.log** - SMTP transport agent private log
* **avgavtasp.log** - SMTP transport agent public log
* **avgasta.log** - Antispam transport agent private log
* **avgastap.log** - Antispam transport agent public log
* **avgcfg.log** – related to the AVG configuration
* **avgcfgex.log** – related to the AVG configuration
* **avgchjw.log** – related to NTFS Journal Caching (e.g. adding file co cache due to valid CertTrusted or failure during saving to cache)
* **avgcore.log** – related to the AVG core
* **avgcsl.log** –related to cloud service (info about uploaded hashes mainly)
* **avgdecider.log** – related to AVG decider
* **avgdiag.log/avgdiagex**.**log** – related to avgdiag
* **avgemc.log –** basic info about avgemc\* (started, stopped, tdistate, ssl emails, certificates, aspam dll)
* **avgexc.log –** related to PUP exception manager (only ERROR messages may be interesting)
* **avgfw.log** – related to AVG Firewall
* **avgfw8db.log** – related to AVG Firewall (related to FW rules created per TrustedDB, IDP whitelist or certificate trusted in AVI )
* **avgfw8u.log** – related to the AVG Firewall (blocked/allowed communication, ARP protection)
* **avgfwui.log** – related to AVG Firewall UI (e.g. opening FW configuration)
* **avgidpagent.log** – related to IDP agent (mainly about functionality like start of IDP processes or detection of malware, former avgidsagent.log)
* **avgidpeh.log** - Early hammer driver log
* **avgldr.log** – related to avi loader
* **avgmail.log** – related to mail plugins
* **avgmf.log -** Minifilterdriver log, info regarding RS when scanning files
* **avgns.log** – related to the Web Shield component (and also email scanning)
* **avgpostinst.log** - related to AVG postinstaller (replacement for FirstRunWizard)
* **avgrkt.log** – related to anti-rootkit
* **avgrs.log** – related to the Resident Shield component
* **avgscan.log** – related to the AVG scans
* **avgsched.log** – related to the Scheduler component
* **avgsps.log** – related to SharePoint scanner
* **avgsrm.log** – related to the Scan Result Manager (it also dependent on the actions performed from the scan, Resident shield, E-mail Scanner)
* **avgsrmac.log** – related to SRM statistics (number of infections detected)
* **avgss.log** - XPL collector
* **avgtdi.log** – related to AVG network filter driver (AVGTDI)
* **avgual.log** – related to the AVG user interface
* **avgui.log** – related to the AVG user interface
* **avguidraw.log** – related to AVG user interface
* **avgupd.log** – related to the update process (virus update)
* **avgvsapi.log** - VSAPI private log
* **avgvsapip.log** - VSAPI public log
* **avgwd.log** – related to the status scanning of AVG components done by the Watchdog component
* **avgwdsvc.log** – related to the AVG Watchdog component
* **commonpriv.log/ commonpub.log** – general logs containing information that cannot be written to another logs (e.g. because logger for this values does not exist on purpose or due to incorrectly configured logger)
* **fixcfg.log** - fixcfg utility log
* **krnlapi**.**log** - Kernel interface log
* **lng.log** - Language interface
* **msi-xxxxxxxx-xxxxxx.log**- related to the installation process (msi installer log)
* **mfa-xxxxxxxx-xxxxxx.log**- related to the installation process (msi front-end application log)
* **vault.log** – related to virus vault
  1. Location of the logs:

**Windows 2000, XP, 2003**

* c:\Documents and Settings\All Users\Application Data\avg2013\Log\
* user
* windows

Installation log:   
c:\Documents and Settings\<user\_name>\Local Settings\Application Data\MFAData\logs\

**Windows Vista, 7, 2008 server**

* c:\ProgramData\avg2013\Log\
* c:\Users\<user>\AppData\Local\Avg2013\log\
* c:\Windows\System32\config\systemprofile\AppData\Local\Avg2013\log\

Installation log:   
c:\Users\<user\_name>\Local Settings\Application Data\MFAData\logs\

1. Encoded & Decoded logs

Logs are encoded by default, so if you wish to see all the info and error messages, it’s necessary to decode them first. Following log file is encoded:

[AVG.MFA.Core] INFO 2010-08-26 06:30:34,093 PC1 PID:704 THID:4036 ID:7E31E31A-D3CF-4270-A532-6A5362FDD68E:18.1089.139344.0 MSG:\*

[AVG.MFA.APP] INFO 2010-08-26 06:30:34,109 PC1 PID:704 THID:4036 ID:0876df7e-4ce9-40da-b5dd-63f1ec9a8b12:116.1089.139344.0 MSG:aAuF91dkT5YU368Ajl4m4y/AyRC1l4TEAweJ6erJ0pzEQssEr/9F13FVEKGWCeU6gQIFSAYkW9dcj9w==

**And here is its decoded form:**

[AVG.MFA.Core] INFO 2010-08-26 06:30:34,093 PC1 PID:704 THID:4036 ID:d:\build\user\sandbox\_2010\_0823\_171149\_avg2012\_Avg13VC9\source\avg13\setupMSI\libs\libmfacore\mfalogging.cpp:18.1089.139344.0 **MSG:Logger initialized**

[AVG.MFA.APP] INFO 2010-08-26 06:30:34,109 PC1 PID:704 THID:4036 ID:d:\build\user\sandbox\_2010\_0823\_171149\_avg2013\_Avg13VC9\source\avg13\setupMSI\avgmfap\application\_runner.cpp:116.1089.139344.0 **MSG:Command line parameters: /pri=4 /session=0 /aspam=0 /dns=0 /appmode=update**

As you can see, MSG texts are hidden in the encrypted logs and the results of actions are hidden as ERRORCODE value. It’s necessary to decrypt the logs first before the analysis, otherwise the analysis are impossible. MSI logs are also encoded in AVG 2013.

Sometimes the analyze of the log files can be difficult. It is recommended to use PS Pad application also with our customization for analysis purposes. See [IKB 700](https://secure.avg.com/ikb.num-700) for further info.

# How to decrypt received log files

To decode the logs, you can the Error Log Decoder in IKB (<http://www.avg.com/ikb>).

You can decode whole CAB archive (avgdiagEX) or you can decode single log files only. After the decoding process, archive with the decoded logs is returned and you can save it on your hard drive for further analysis.

In decoded files, there still can be some error codes, for example: 0x80070005. You can decode such error code

* on LabdoraNG at <http://labdorang.cz.avg.com> (**Error Codes Translator** link), or
* in IKB (**Error codes**).

The error code from the above mentioned example means *Access Denied*.

1. Structure of the encrypted log records

Each line contains of the following information:

* Name of the logger which sent the message to be logged
* Severity
* Date of the event
* Time of the event
* Name of the computer
* Process ID
* Thread ID
* ID and source of the logged component (useless for normal analysis)
* Message text

The only important records are Name of the logger, Severity, date and time, Message text. All AVG logs have this structure.

## Logged component (Name of the logger)

In the beginning of every log record, source of the record is displayed in the square brackets. For example:

* **[AVG.MFA.APP]** – record is created by MsiFrontendApplication (used by update and installation)
* **[AVG.RS]** – record is created by the AVG Resident Shield
* **[AVG.RS.Core]** – record is created by the scanning process of the resident shield
* **[AVG.SCHED]** – record is created by the Scheduler component
* **[AVG.SCHED.Task1]** – record is created by specific scheduled task
* **[AVG.AVGFW.FwSrv]** – record is created by AVG Firewall
* **[AVG.Core]** – record created by AVG Core
* **[AVG.ADMINCLI]** – record created by the Remote administration component
* **[AVG.SCAN]** – record created by the AVG scan

## Severity

INFO – this record is just an information about AVG behavior. For example about successful update, detected proxy, scanned file, etc.

[AVG.MFA.APP] INFO 2010-08-26 06:30:34,109 PC1 PID:704 THID:4036 ID:d:\build\user\sandbox\_2010\_0823\_171149\_avg2013\_Avg13VC9\source\avg13\setupMSI\avgmfap\application\_runner.cpp:116.1089.139344.0 **MSG:Command line parameters:** /**pri=4** /**session=0** /**aspam=0** /**dns=0** /**appmode=update**

* it means, that the update was started with the mentioned parameters

DEBUG – this record is an output of the debug logging. It records all actions done by AVG. For example:

[AVG.MFA.UPD] DEBUG 2010-08-26 06:32:15,234 PC1 PID:624 THID:4004 ID:d:\build\user\sandbox\_2010\_0820\_101948\_avg2013\_Avg13VC9\source\avg13\setupMSI\libs\libupd\mfaupdate.cpp:351.1086.139057.0 **MSG:Last working IP url: 212.96.161.238|http://update.avg.com/beta/10/update/**

* it specifies the last working IP – used by protection mechanism

DETAIL – it’s also output of the debug logging. For example:

[AVG.MFA.UPD] DETAIL 2010-08-26 06:30:35,296 PC1 PID:704 THID:244 ID:d:\build\user\sandbox\_2010\_0823\_171149\_avg2013\_Avg13VC9\source\avg13\setupMSI\libs\libinet\downloader.cpp:921.1089.139344.0 **MSG:Download of avg13infoavi.ctf started**

* it means, that update started downloading of a file

WARN – AVG component has non-critical problem. For example:

[AVG.CSL] WARN 2010-08-16 14:23:06,553 PC1 PID:696 THID:3804 ID:d:\nightly\sandbox\_avg13\_vc9\_trunk\source\avg13\common\avgcsl\avgcloudclient.cpp:996.1081.138368.0 **MSG:HEARTBEAT disabled by throttling mechanism.**

* Heartbeat is feature checking for new updates on a specific occasion. It is related to new feature cloud services. From the log you can read that this feature was disabled by throttling mechanism which is also part of cloud services mechanism. This is only a warning as this does not affect the proper functionality.

ERROR – some AVG component had serious problem. For example:

[AVG.MFA.UPD] ERROR 2010-08-26 06:32:20,703 PC1 PID:624 THID:4072 ID:d:\build\user\sandbox\_2010\_0823\_171149\_avg2013\_Avg13VC9\source\avg13\setupMSI\libs\libinet\downloader.cpp:1230.1089.139344.0 **MSG:Curl error code: 6, (Couldn't resolve host name)**

* it means, that AVG update process had problem to resolve update server name to the IP address (problem with DNS)

## Date and time

The date and time is displayed in the GMT. Exception are MSI logs (msi install logs and avgupdm.log). Time in these logs is current user time shown on computer. Time in sys\_info.xml is in GMT.

## Message text

It’s decoded text message initiated by the MSG text. For example:

[AVG.MFA.Core] INFO 2010-08-26 06:30:34,093 PC1 PID:704 THID:4036 ID:d:\build\user\sandbox\_2010\_0823\_171149\_avg2013\_Avg13VC9\source\avg13\setupMSI\libs\libmfacore\mfalogging.cpp:18.1089.139344.0 **MSG:Logger initialized**

* this particular record informs you that the update was started

1. How to change the logging level

* To change the logging level, it is necessary to modify the corresponding configuration file located here:

|  |  |  |
| --- | --- | --- |
|  | **Windows XP** | **Vista** |
| **\*.cfg file** | c:\Documents and Settings\All Users\Application Data\avg2013\Log\ | c:\Program Data\avg2013\Log\ |

* If we need to enable debug logging on customer’s computer, we always send him/her modified configuration file(s).
* To enable debug logging, we need to change **Severity** value in corresponding log file.
* In order to apply the change of logging level, **computer restart is required**.
* Default state:

|  |
| --- |
| [AVG.WD]  Type = Logger  Severity = Info  Additive = false  AddAppenders = wdFileAppender |

* Debug logging enabled:

|  |
| --- |
| [AVG.WD]  Type = Logger  **Severity = Debug**  Additive = false  AddAppenders = wdFileAppender |

1. Numbering and rolling of log files

* By default, size of one logging files is set to 1 024 000 bytes (approx. 1 MB). Size can be changed in corresponding configuration log in:  
  c:\Documents and Settings\All Users\Application Data\avg2013\Log\
* Default state:

|  |
| --- |
| [coreFileAppender]  Type = Appender  AppenderType = RollingFile  **Size = 1024000**  FilesCount = 10  FileName = avgcore.log |

* Size of the log doubled:

|  |
| --- |
| [coreFileAppender]  Type = Appender  AppenderType = RollingFile  **Size = 2048000**  FilesCount = 10  FileName = avgcore.log |

* When the log file size reaches the defined limit, the original log file is renamed – number is added as an extension. For example:  
  **avgrs.log** -> **avgrs.log.1**
* Maximum number of older log files is defined by **FilesCount** parameter. Default value is **10**.
* Default state (avgrs.log – avgrs.log.9):

|  |
| --- |
| [coreFileAppender]  Type = Appender  AppenderType = RollingFile  Size = 1024000  **FilesCount = 10**  FileName = avgcore.log |

* Number of log files extended to 20 (avgrs.log – avg.log.19):

|  |
| --- |
| [coreFileAppender]  Type = Appender  AppenderType = RollingFile  Size = 1024000  **FilesCount = 20**  FileName = avgcore.log |

* Extending of log files size can be useful while troubleshooting problems, where we need data from longer period of time (e.g. AVG scan – enumeration of all scanned files).

1. Translation of error codes

* While analyzing log files, you will most probably encounter some error codes (e.g. 0x80070005). To translate the error code, use the translator available at:
  + <http://labdorang.cz.avg.com/LabServices/ErrorCodes>   
    (LabdoraNG-> Error Codes Translator)
  + <https://secure.avg.com/230619>   
    (IKB -> Error codes)

# Examples of some logs

## Personal e-mail scanner

Most of email handling functions (connection to remote POP3/SMTP/IMAP servers, scanning for viruses) is now provided by avgns and avgtdi. Avgtdi forwards data from ports used for email connections to avgns and avgns can provide part of these data to emc. Avgemc is now mainly responsible for SSL connections (encoding, decoding), antispam scan, modifications of email body (certification text, modification of subject and conversion to html/rtf/plain)

For analysis of email issues it is now necessary to have logs from avgemc as well as from avgns - avgemc.log and avgns.log. Both logs are encoded.

avgemc.log

This record shows correct start of avgemc[x/a].exe:

MSG:EMS command server starting ...

MSG:Config manager initialized

MSG:RefreshEmsState, state: 40, TDIState: 1

MSG:EMS command server started

MSG:EMS starting ...

MSG:Proxy server configuration added (port: 6e)

MSG:Proxy server configuration added (port: 19)

MSG:Proxy server configuration added (port: 8f)

MSG:RefreshEmsState, state: 51, TDIState: 1

MSG:RefreshEmsState, state: 100, TDIState: 1

MSG:EMS started

avgns.log

This record shows corect download of email. Debug log would show even more info:

MSG:connid:108 POP3: Command: USER

MSG:connid:108 POP3: Response: +OK

MSG:connid:108 POP3: Command: PASS

MSG:connid:108 POP3: Response: +OK Logged in.

MSG:connid:108 POP3: Command: STAT

MSG:connid:108 POP3: Response: +OK 1 1156

MSG:connid:108 POP3: Command: LIST

MSG:connid:108 POP3: Response: +OK 1 messages:

MSG:connid:108 POP3: Command: RETR

MSG:connid:108 POP3: Response: +OK 1156 octets

MSG:connid:108 POP3: Command: DELE

MSG:connid:108 POP3: Response: +OK Marked to be deleted.

MSG:connid:108 POP3: Command: QUIT

MSG:connid:108 POP3: Response: +OK Logging out, messages deleted.

## 8.2 Install log

AVG 2013 is using MSI installer for installation and program update. During each installation, three logs are created. One is related to AVG installer avgmfap[a/x].exe which handles unpacking of installation package and input parameters, second is MSI installer log itself. Third log belongs to Visual C++ Redistributable package. This can be useful, when installation is failing during VC++ redist installation. As an example we can have:

**mfa-20120905-105817.log**

**msi-20120905-105817.log**

**r86-20120905-110229.log**

Structure of install logs is quite complicated. However you are still able to find a few important information in logs. For manual analysis, PSPad is the best. You can use it to list only rows which contain specific word:

### parameters used during installation - mfa\*.log

search for **MSG:Command line parameters:**

[AVG.MFA.APP] INFO 2010-08-24 11:50:33,029 MATEJEK-WXP-X32 PID:1316 THID:1348 MSG:Command line parameters: .\avgmfapx.exe /LicenseKey="9ISMB-J6T4C-6PVGW-M0XFR-TREZZ-33KFT-XXXX"

### license number used for installation - mfa\*.log

search for **LicenseKey=**

[AVG.MFA.APP] INFO 2010-08-24 07:18:57,828 PC1 PID:412 THID:1416 MSG:Variable storage dump before run:||ApplicationDirectory=C:\DOCUME~1\ADMINI~1\LOCALS~1\Temp\7zS1.tmp||ApplicationPath=C:\DOCUME~1\ADMINI~1\LOCALS~1\Temp\7zS1.tmp\avgmfapx.exe||AppMode=setup||BrandId=1||ConfigFilePath=C:\DOCUME~1\ADMINI~1\LOCALS~1\Temp\7zS1.tmp\mfaconf.txt||IsAdminProduct=0||IsDowngrade=0||IsUpgradeMode=0||LicenseKey=**9BETA-R8KPC-TQNKP-NPW7R-L440X-KPE7J-K4GP**||LNGCanUseFileMapping=1||LoggerPath=C:\DOCUME~1\ADMINI~1\LOCALS~1\Temp\MFAData\logs||MfaMode=6||MsiLogFilePath=C:\DOCUME~1\ADMINI~1\LOCALS~1\Temp\MFAData\logs\msi-20100824-071857.log||PackagePath=C:\DOCUME~1\ADMINI~1\LOCALS~1\Temp\MFAData\pack||PackageType=IS||ProductId=4||ProductPkgId=AVG||ProductType=IS||TemporaryPath=C:\DOCUME~1\ADMINI~1\LOCALS~1\Temp\MFAData||UILevel=normal||UILevelType=1||

### version of AVG installation package - msi\*.log

search for **ProdVer=**

MSI (c) (9C:88) [09:19:01:375]: PROPERTY CHANGE: Adding PKGINFO property. Its value is 'ProdVer=**2013.0.1831**, ProdType=IS\*, Installed=, WinVer=501\*, WinBuild=2600, SP=3, WinType=1'.

### user who launched the installation - msi\*.log

search for **LogonUser**. Although you cannot see priviledges, you can find at least a ccount name:

Property(S): LogonUser = **Administrator**

### installation directory - msi\*.log

search for **Property(S): AVGVERFOLDER**

Property(S): AVGVERFOLDER = C:\Program Files\AVG\AVG2013\

components selected for installation - msi\*.log

filter rows with **Feature:** and from this filter only rows which contain **MSI (s)** . Similar rows should be shown:

* *clean installation (all components selected)*

MSI (s) (B8:58) [09:21:22:515]: Feature: fea\_Antivir; Installed: Absent; Request: Local; Action: Local

MSI (s) (B8:58) [09:21:22:531]: Feature: DEPfea\_ResShld.fea\_Antivir; Installed: Absent; Request: Local; Action: Local

MSI (s) (B8:58) [09:21:22:531]: Feature: fea\_ResShld; Installed: Absent; Request: Local; Action: Local

MSI (s) (B8:58) [09:21:22:515]: Feature: fea\_IDP; Installed: Absent; Request: Local; Action: Local

* *clean installation (IDP unchecked, rest enabled)*

MSI (s) (E0:98) [13:53:25:310]: Feature: fea\_Antivir; Installed: Absent; Request: Local; Action: Local

MSI (s) (E0:98) [13:53:25:310]: Feature: DEPfea\_ResShld.fea\_Antivir; Installed: Absent; Request: Local; Action: Local

MSI (s) (E0:98) [13:53:25:310]: Feature: fea\_ResShld; Installed: Absent; Request: Local; Action: Local

MSI (s) (E0:98) [13:53:25:310]: Feature: fea\_IDP; Installed: Absent; Request: Null; Action: Null

* *repair installation (all components selected)*

MSI (s) (AC:B4) [11:01:24:605]: Feature: fea\_Antivir; Installed: Local; Request: Reinstall; Action: Reinstall

MSI (s) (AC:B4) [11:01:24:605]: Feature: DEPfea\_ResShld.fea\_Antivir; Installed: Local; Request: Reinstall; Action: Reinstall

MSI (s) (AC:B4) [11:01:24:605]: Feature: fea\_ResShld; Installed: Local; Request: Reinstall; Action: Reinstall

MSI (s) (AC:B4) [11:01:24:605]: Feature: fea\_IDP; Installed: Local; Request: Reinstall; Action: Reinstall

* *add or remove components (AVG already present, IDP selected for removal, rest enabled)*

MSI (s) (70:3C) [15:02:11:916]: Feature: DEPfea\_ResShld.fea\_Antivir; Installed: Local; Request: Null; Action: Null

MSI (s) (70:3C) [15:02:11:916]: Feature: fea\_ResShld; Installed: Local; Request: Local; Action: Local

MSI (s) (70:3C) [15:02:11:916]: Feature: fea\_IDP; Installed: Local; Request: Absent; Action: Absent

As you can see above, we can also find some dependencies here. Dependencies only say that some component is dependent on some specific CAB file. As long as some component requires the CAB file, the CAB file will remain in temporary folder after installation. Otherwise the CAB is removed.

Also you can find the current state of component:

Installed: Absent - not present

Installed: Local - present

Request: Local - install it

Request: Reinstall - reinstall it

Request: Null - do nothing

Request: Absent - remove it

Action: Local - install it

Action: Reinstall - install it

Action: Null - do nothing

Action: Absent - remove it

From this you can also conclude if AVG was already installed

error on which installation failed

Note that most of failures will show simillar error code "0xC0070643" in UI. However cause of failure differs! To find the exact cause of failure, we need to check the MSI install log msi-\*-\*.log **.**

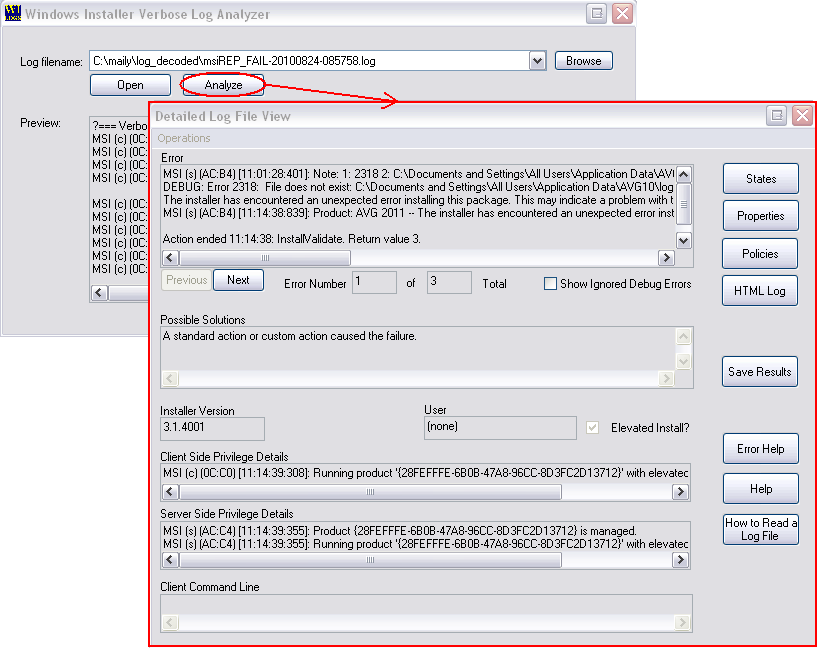
The important thing you must know about MSI installation is that installation stops and rollback is done imediatelly after first failure found. This means that if you want to find the error in install log, you should read the log from the bottom to top. Top of the log is usually full of uninteresting information. The last (first from the top) error you find this way is the one that caused error.

As we now know that first error starts rollback action, we could also say that if user has 2 problems blocking the installation on his computer, following will happen:

1. User starts installation which fails on problem1
2. In the MSI log we find problem1 and recommend fix (problem 2 will not be shown in log as installation stopped right after problem1)
3. User performes proposed fix and installation now fails on problem2
4. In new MSI log we find problem2 and recommend fix
5. Now the installation result is success

The fastest and reliable way how to find cause of installation us to use Microsoft utility **wilogutl.exe** or **labdora** portal:

<http://labdorang.cz.avg.com/LabServices/MsiLog>



The picture above clearly states where the problem is (see part ERROR).

First important lines can be cut out:

You will also encouter lots of installation errors where labdora/wilogutl does not show all the lines related to error. In such a case you need to open MSI log manually and search for the reason of failure:

*WiLogUtl reports:*

1: AvgMsi: Read multistring failed: -536805315 (0xe001003d)

1: AvgMsi: End of TestRegSetReg, result: -536805315 (0xe001003d)

1: AvgMsi: TRSR res: -536805315 (0xe001003d)

1: AvgMsi: End of MainMsiFun, result: -536805315 (0xe001003d)

CustomAction CallSetBootExecChsv returned actual error code 1603 (note this may not be 100% accurate if translation happened inside sandbox)

Action ended 0:23:58: InstallFinalize. Return value 3.

This does not give us enough information about failure. Therefore we will open the MSI install log and search for some of mentioned lines. Then we need to check a few rows above to understand the problem. We will find:

MSI (s) (08:18) [00:23:58:432]: Executing op: ActionStart(Name=CallSetBootExecChsv,,)

MSI (s) (08:18) [00:23:58:463]: Executing op: CustomActionSchedule(Action=CallSetBootExecChsv,ActionType=3073,Source=BinaryData,Target=MainMsiFun,CustomActionData=TRSR;**HKLM;SYSTEM\CurrentControlSet\Control\Session Manager**;**BootExecute**;avgchsva.exe;C:\Program Files (x86)\AVG\AVG2013\avgchsva.exe; /sync)

MSI (s) (08:70) [00:23:58:510]: Invoking remote custom action. DLL: C:\Windows\Installer\MSICA5A.tmp, Entrypoint: MainMsiFun

1: AvgMsi: Begin of MainMsiFun, hInstall: 3382 (0x00000d36)

1: AvgMsi: Admin rights: : 1 (0x00000001)

1: AvgMsi: SYSTEM

1: AvgMsi: Begin of TestRegSetReg, hInstall: 3382 (0x00000d36)

1: AvgMsi: Read multistring failed: -536805315 (**0xe001003d**)

1: AvgMsi: End of TestRegSetReg, result: -536805315 (0xe001003d)

1: AvgMsi: TRSR res: -536805315 (0xe001003d)

1: AvgMsi: End of MainMsiFun, result: -536805315 (0xe001003d)

CustomAction CallSetBootExecChsv returned actual error code 1603 (note this may not be 100% accurate if translation happened inside sandbox)

Action ended 0:23:58: InstallFinalize. Return value 3.

Now this gives us a good understanding to error. Installation failed on adding value to reg key BootExecute – the key does not exist (0xe001003d). Workaround is to create the key manually.

Errors can be hidden deeper in log

It may happen that wilogutl/labdora targets some error and looking at a few lines above does not help as they are not error related. In that case, it is necessary to continue searching in the logs from the bottom to the top and find the latest error shown (first error from the bottom). You can also try to search for „**CustomActionRollback**“ from the top. If present, this value will indicate when exactly rollback started – right above this you can find some failed action.

WiLogUtl/Labdora says:

MSI (s) (B8:48) [18:04:47:673]: Destroying RemoteAPI object.

MSI (s) (B8:28) [18:04:47:673]: Custom Action Manager thread ending.

MSI (c) (0C:74) [18:04:47:688]: Back from server. Return value: 1603

MSI (c) (0C:74) [18:04:47:688]: Decrementing counter to disable shutdown. If counter >= 0, shutdown will be denied. Counter after decrement: -1

MSI (c) (0C:74) [18:04:47:688]: PROPERTY CHANGE: Deleting SECONDSEQUENCE property. Its current value is '1'.

Action ended 18:04:47: ExecuteAction. Return value 3.

Right above this record, there is only some information not related to error:

Property(S): fea\_TuneUp\_Vis = 0

Property(S): LIC\_OK = 1

Property(S): SOURCEDIR = C:\DOKUME~1\ADMINI~1\LOKALE~1\Temp\7zS55.tmp\

Property(S): SourcedirProduct = {8FB6AF1C-7B7B-42F9-BAAF-7592AC9819E6}

Property(S): FILESYSTEMTYPE = NTFS

Property(S): ProductToBeRegistered = 1

MSI (s) (B8:F0) [18:04:47:579]: MainEngineThread is returning 1603

In this case we need to go deeper in the log. The easiest way is to search for the first occurance of **CustomActionRollback**. Only if searching fails, we need to start checking the log manually from the bottom. In our example, rollback was present:

MSI (s) (B8:F0) [18:04:44:313]: Executing op: ActionStart(Name=CallStopW3SVC\_In,,)

MSI (s) (B8:F0) [18:04:44:329]: Executing op: CustomActionSchedule(Action=CallStopW3SVC\_In,ActionType=3073,Source=BinaryData,Target=MainMsiFun,CustomActionData=svcFunEx;W3SVC;O;;;C:\DOKUME~1\ADMINI~1\LOKALE~1\Temp\)

MSI (s) (B8:CC) [18:04:44:345]: Invoking remote custom action. DLL: C:\WINDOWS\Installer\MSI97.tmp, Entrypoint: MainMsiFun

1: AvgMsi: Begin of MainMsiFun, hInstall: 2796 (0x00000aec)

1: AvgMsi: Admin rights: : 1 (0x00000001)

1: AvgMsi: SYSTEM

1: AvgMsi: Begin of ServiceFunEx, hInstall: 2796 (0x00000aec)

1: AvgMsi: Extended Service Maintenance res: -1073282012 (0xc0070424)

1: AvgMsi: End of MainMsiFun, result: -1073282012 (0xc0070424)

Aktion beendet um 18:04:45: InstallFinalize. Rückgabewert 3.

MSI (s) (B8:F0) [18:04:45:267]: User policy value 'DisableRollback' is 0

MSI (s) (B8:F0) [18:04:45:267]: Machine policy value 'DisableRollback' is 0

MSI (s) (B8:F0) [18:04:45:282]: Executing op: Header(Signature=1397708873,Version=301,Timestamp=1027444881,LangId=1033,Platform=0,ScriptType=2,ScriptMajorVersion=21,ScriptMinorVersion=4,ScriptAttributes=1)

MSI (s) (B8:F0) [18:04:45:282]: Executing op: DialogInfo(Type=0,Argument=1033)

MSI (s) (B8:F0) [18:04:45:282]: Executing op: DialogInfo(Type=1,Argument=AVG 2011)

MSI (s) (B8:F0) [18:04:45:282]: Executing op: RollbackInfo(,RollbackAction=Rollback,RollbackDescription=Aktion wird rückgängig gemacht:,RollbackTemplate=[1],CleanupAction=RollbackCleanup,CleanupDescription=Sicherungsdateien werden entfernt,CleanupTemplate=Datei: [1])

MSI (s) (B8:F0) [18:04:45:282]: Executing op: ActionStart(Name=CallStopW3SVC\_In,,)

MSI (s) (B8:F0) [18:04:45:282]: Executing op: ProductInfo(ProductKey={8FB6AF1C-7B7B-42F9-BAAF-7592AC9819E6},ProductName=AVG 2011,PackageName=AVGx86.msi,Language=1033,Version=167773280,Assignment=1,ObsoleteArg=0,,,PackageCode={86ACE36D-F4C7-4656-B8F9-876AC67222A3},,,InstanceType=0,LUASetting=0,RemoteURTInstalls=0)

MSI (s) (B8:F0) [18:04:45:282]: Executing op: ActionStart(Name=CallStopExchgIS\_InRB,,)

MSI (s) (B8:F0) [18:04:45:282]: Executing op: **CustomActionRollback**(Action=CallStopExchgIS\_InRB,ActionType=3329,Source=BinaryData,Target=MainMsiFun,CustomActionData=svcFunEx;MSExchangeIS;A;;;C:\DOKUME~1\ADMINI~1\LOKALE~1\Temp\)

CustomtActionRollback can be seen in the end of the log. Searching for the firest error above we can find, that problem occured with stopping W3SVC. Error code 0xc0070424 can be translated as ERROR\_SERVICE\_DOES\_NOT\_EXIST. W3SVC is related to SharePoint plugin. AVG detects, whether registry for SharePoint exists. If it does, plugin for SharePoint is offered for installation. In this case, registry existed but service does not – it is problem on customers PC. Failure is not in AVG.

## 8.3 Update log

Update and setup is handled by avgmfap[x/a].exe. Virus update is performed strictly via this process and is recorded into **avgupd.log**.Program update is done in a several steps - at first avgmfap[x/a] downloads update bin files and then it calls MSI installer to perform the program update. Download of bins is recorded to **avgupd.log**, progress of applying program update is recorded to **avgupdm.log** – this is MSI log file and it can be again parsed on portal<http://labdorang.cz.avg.com/LabServices/MsiLog> .

Example of virus update in **avgupd.log**:

update always starts with:

MSG:Logger initialized

parameters used for update:

MSG:Command line parameters: /pri=4 /session=0 /aspam=0 /dns=0 /appmode=update

bins selected for download can be found as:

MSG:Selected bin: u10iavi3091u3090av.bin

exiting update process:

MSG:Unregistered from watchdog.

# AVG diagnostic packages

* 1. AVGDIAGX output

For details about the utility, see **[IKB 1458: AvgDiagEx functionality and features](https://secure.avg.com/ikb.num-1458).**

Besides logging files, AVG diagnostic utility also gathers xml reports that contains general information about system and might be very usefull for troubleshooting:

* **autoruns.xml** – Usefull for solving infection issues or conflicts with other software. Contains registry keys that are likely to be affected by infection:
* Enumeration of HKLM Autoruns
* Enumeration of HKCU Autoruns
* Enumeration of HKLM Image File Execution Options
* Enumeration of HKLM Session Manager - BootExecute
* Enumeration of HKLM Winlogon
* Enumeration of HKLM-HKCU System Policies
* Enumeration of HKLM-HKCU Explorer Policies
* Enumeration of HKLM Session Manager
* Enumeration of HKLM TCPIP Parameters
* Enumeration of HKLM-HKCU (User) Shell Folders
* Enumeration of HKCU Internet Settings
* Enumeration of HKLM-HKCU Internet Explorer - Main
* Enumeration of HKLM Shell Execute Hooks
* Enumeration of HKLM BHO
* Enumeration of HKCR Executables Extensions Shell Open Command
* **avg\_info.xml** – basic information about AVG (license number, virus database) and about AVG files:
* AVG License Info
* AVG Basic Info
* AVG Core Components
* AVG Executables
* AVG Libraries
* AVG IDS Components
* E-mail clients
* Web browsers
* AVG for Exchange
* **env\_info.xml** – information related to installed applications, running services, devices and settings:
* Enumeration of processes and their modules
* Enumeration of HKLM-HKCU AVG key
* Enumeration of HKLM Services - Full List
* Enumeration of Devices
* Enumeration of Services
* Enumeration of Uninstall strings
* Enumeration of Patches
* Enumeration of HKLM Autoruns
* Enumeration of HKCU Autoruns
* Enumeration of Browser Helper Objects
* **sys\_info.xml** –system information
* Basic System Information
* Current User Access Rights
* Enumeration of Users Access Rights
* Enumeration of Discs
* Environment Variables
* System specific files

## FWDIAG output

FWDIAG utility creates utility package packed in the archive. The archive has following structure:

* **CFGS folder –** include CFG files for most AVG components
* **CFGSall folder –** include CFG files from folder cfgall (e.f. fw.cfg)
* **Infs folder –** contains all INF files from computer (e.g. oem1.inf)
* **LOGS folder –** include all AVG log files
* **appdump.txt –** list of the installed applications. It’s export from the registry key [HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall]
* **arpdump.txt** - arp table (cmd -> arp –a)
* **avg8dump.txt** – list of AVG2013 keys in HKLM\Software
* **ccsdump.txt** – export of registry keys (hardware profiles, services…)
* **condump.txt** – it the list of the UPD and TCP connections
* **drvdump.txt** – list of drivers on the system and their states (driverquery /v)
* **ipcdump.txt** – output from ipconfig /all command
* **iprdump.txt** – locally saved route table
* **lspdump.txt** – information about Winsock2 (export from the registry key [HKLM\SYSTEM\CurrentControlSet\Services\WinSock2]
* **netdump.txt** – it the list of the UPD and TCP connections
* **procdmp.txt** – list of the running processes
* **svcdump.txt** – AVG Firewall statistics
* **sysdump.txt** – info about operating system, logged user and mapped drives
* **tdidump.txt** – information about TDI driver
* **trfdump.txt** – contains all communication from last report to M-trap. Can be used for analysis of blocked communication (wrong rules)
* **verdump.txt** – list of the versions of AVG executables

# AVG issues and important log files

Please bear in mind that the following log files are the most used logs for analysis in the mentioned situation. It doesn’t mean, that the rest of the logs is useless for this situation.

## Update problems

Information about this problem can be found in the log file(s):

* avgupd.log

## Problem with scheduled tasks

Information about problems with this component can be found in the log file(s):

* avgsched.log

## Problems with Resident Shield

Information about problems with this component can be found in the log file(s):

* avgrs.log
* avgldr.log (if RS is not running at all)

## Problems with AVG scans

Information about problems with this component can be found in the log file(s):

* avgrkt.log (if it is related to anti-rootkit scan)
* avgscan.log

## Problems with Online Shield (or LinkScanner)

Information about problems with this component can be found in the log file(s):

* avgns.log
* avgldr.log (if OS is not running at all)

## **Problems with AVG gui (crashes, unexpected** moving, closing, etc)

Information about problems with this component can be found in the log file(s):

* avgui.log
* avgcfg.log (if there is any problem with saving the configuration settings)

## Installation problem

Information about problems with this component can be found in the log file(s):

* mfa-xxxxxxxx-xxxxxx.log
* msi-xxxxxxxx-xxxxxx.log

## Problem with E-mail scanner

Information about problems with this component can be found in the log file(s):

* avgemc.log
* avgns.log (ns needs to be active to have EMC running, also contains info about email scanning)

## Problems with Identity Protection component

Information about problems with this component can be found in the log file(s):

* avgidpagent.log is related to main IDP functionality
* avgtray\_idp\_\*.log, avgui\_idp\_\*.log are related to IDP implmentation in AVG (e.g. UI, tray)
* history.xml (records about detected samples are there)

## Problems with Anti-Spam

By default, Anti-Spam is not logging on a client station. You can enable debug logging by modification of „spamcatcher.conf“. You should also gather avgemc and avgns logs if user has personal e-mail scanner.

## Firewall issues and important log files

Please bear in mind that the following log files are the most used logs for analysis in the mentioned situation. It doesn’t mean, that the rest of the logs is useless for this situation.

### Firewall is blocking some application

Following files from the FWDIAG output are important:

* **fw.cfg** – you can import firewall configuration and review users setting
* **tdidump.txt** – you can see what rules were in effect at time, when fwdiag was gathered (e.g. not all rules related to svchost.exe are visible in fw.cfg)
* **avgfw8db.log** – you can see, whether the application was set as blocked per TrustedDb or Sana (Sana can determine Allow or Block rule; TrustedDB can determine Allow, block or Custom and CertTrusted can be used only for allow)
* **avgfw8u.log** – here, you can see the history of the blocked applications and system services and you can check whether the application was really blocked and what IP addresses + ports were used. You can also find „BlockedIDS“ records which are related to portscan and ARP protection – some communication could be blocked due to getting on blacklist.
* **appdump.txt** – here, you can check whether the user uses any other firewall, content filtering application and other applications which may cause this problem
* **svcdump.txt –** here, you can check what profile is activated
* **verdump.txt** – here, you can check whether the user uses the latest AVG Firewall version
* **avgfw.log** – you can find whether the profile wasn’t switched lately

### Firewall failed to start

Following files from the FWDIAG output are important:

* **appdump.txt** – here, you can check if there are any conflict applications installed on the computer
* **cssdump.txt** – check that AVG FW services startup is not set to disabled (0x4)
* **fw.cfg** – you can import firewall configuration to see configuration is not corrupted
* **ipcdump.txt** – you can check how many internet connections (or adapters) are installed on the computer
* **sysdump.txt** – you can find out what operating system is affected
* **procdmp.txt** – it’s also recommended to check whether there are any suspicious or conflicting processes running
* **avgfw.log** – try to find an error messages and compare them with the known bugs in the bugzilla

### Firewall crashed or causes crash

Following files from the FWDIAG output are important:

* **appdump.txt** – here, you can check if there are any conflict applications installed on the computer
* **sysdump.txt** – you can find out what operating system is affected
* **procdmp.txt** – it’s also recommended to check whether there are any suspicious or conflicting processes running
* **avgfw.log** – try to find an error messages related to the crash and compare them with the known bugs in the bugzilla

And of course analysis of the **DUMP** files is necessary too.

### Firewall is not blocking any specific communication

Following files from the FWDIAG output are important:

* **fw.cfg** – you can import firewall configuration and review users setting
* **tdidump.txt** – you can see what rules were in effect at time, when fwdiag was gathered (e.g. not all rules related to svchost.exe are visible in fw.cfg)
* **avgfw8db.log** – you can see, whether the application was set as allowed per CertTrust, TrustedDb or SanaWhitelist
* **avgfw8u.log** – you can check whether the application was really allowed or not
* **ipcdump.txt** – you can check how many internet connections (or adapters) are installed on the computer and determine what adapter is affected
* **procdmp.txt** – here, you can find out PID of the problematic application or process
* **netdump.txt** – active connections, it may help you to find which other applications may be related to the connection
* **avgfw.log** – you can find whether the profile wasn’t switched lately

1. Configuration
   1. General information

* Configuration file for any AVG component is not created until the settings of such components are changed. If the configuration does not exist, settings of the component have not been changed.
* To import the configuration file, it is only needed to copy it to respective folder (see below). Usually, it is not needed to restart any AVG service or component.
  1. List of AVG configuration files
* **%ALLUSERSPROFILE%\Application Data\avg2013\Cfg\**
  + **admin.cfg** - DataCenter connection
  + **advisor**.cfg – Advisor settings (browser proces monitoring, torrent monitoring, wifi-guard, …)
  + **ameh.cfg** - Alert Manager configuration
  + **amrules.cfg** - Alert Manager rules
  + **apiall**.**cfg** – related to AVG API usage
  + **aspam.cfg** - Anti-Spam configuration
  + **aspmlist.cfg** - White/Black Lists configuration
  + **awacs**.**cfg** – related to campaigns and endpoint user segmentation
  + **cachesrv.cfg** – CacheServer configuration
  + **changecfgreg.cfg** – affects possibility to change cfg and registry by avi update
  + **csl**.cfg – Cloud service configuration
  + **dav**.**cfg** – Cfg for DisableProtection function
  + **emssrv.cfg** - EMC configuration - automatic and manual servers
  + **erd**.**cfg** – cfg related to downgrade to free function
  + **except.cfg** - PUP exceptions
  + **falsealarm.cfg** – related to automatic restoration of falsely detected files from virus vault. For details see [IKB 1647](https://secure.avg.com/ikb.num-1647)
  + **fw.cfg** - Firewall configuration (binary)
  + **hlog.cfg** - History Logger configuration, *not used for logging in 2013 release*
  + **idp**.**cfg, idp2.cfg** – IDP configuration
  + **idpallow.cfg** – IDP allow list and information about allowed items
  + **idp\_update.cfg** – related to IDP update (timeout settings, next query time, …)
  + **isaserver.cfg** – ISA server configuration.
  + **krnl.cfg** - Main program settings
  + **krnlall.cfg** - additional main program settings
  + **mail.cfg** - E-mail scanner - logging, progress, queue...
  + **mailsrv.cfg** – configuration of transport agent (Exchange)
  + **mailsrvvsapi.cfg** – configuration of VSAPI scanner (Exchange)
  + **malrep.cfg** – malware reporting (prelevance reporting)
  + **oop.cfg** – related to out-of-process scanning and healing
  + **outlook.cfg** – MSO bar settings, internal Outlook junk filter
  + **pctuneupall**.**cfg –**PC Analyzer (date of last analysis, detection of standalone PC TuneUp)
  + **popup**.**cfg** – related to notifications/campaigns
  + **restart.cfg** – related to restart notification and postponing
  + **rsexcludes.cfg** – Resident shield exceptions configuration file
  + **scan.cfg** - Scan configuration (manual, scheduled, anti-rootkit, gadget scan, …)
  + **sched.cfg** - Scheduler configuration
  + **setup.cfg** - Setup configuration - e.g. what components have been installed
  + **spsrv.cfg** – SharePoint scanner configuration
  + **srmall.cfg** – contains only one option which is set automatically by AVG (if AVG requires restart to remove some infection then it is set to True to change state of the infection in the scan results after restart – then it is set to False again)
  + **station.cfg** – station license number and registration details
  + **tb**.**cfg** – toolbar configuration file, *not used for logging in 2013 release*
  + **update.cfg** - update configuration
  + **updateall.cfg** – additional update settings (including license number change by server)
  + **updatecomps.cfg** - Information about component build numbers, virus db versions and required restart after update
  + **upgrade**.**cfg** – related to major version upgrade
  + **user.cfg** - user settings - notifications, ignored error state of components, first scan dialog etc
  + **userall**.**cfg** – additional user settings
  + **userawacs.cfg** – information on active campaigns, installed applications and app offers
  + **usergui**.**cfg** – various GUI settings (sounds, language, pop-ups for Analyzer etc)
  + **userguistate**.**cfg** – related to report items and baners
  + **wd**.**cfg** – related to Boot Accelerator and update process
* **%ALLUSERSPROFILE%\Application Data\avg2013\Antispam\**
  + **spamcatcher.conf** - Additional Anti-Spam (MailShell) configuration – logging
* **%allusersprofile%\MFAData\logs\**