A MITEL
PRODUCT
GUIDE

Mitel OpenScape Personal Edition

Trace Guide

Administrator Guide

09/2024

A31003-G2570-M100-02-76A9

Notices

Trademarks

Contents

1 History of change

Version	Date	Changes
2.0	11/06/2014	New Trace Guide document for GVS and BLS uses.
2.01	12/11/2014	Updated formatting of Trace Guide document.
2.02	12/12/2014	Updated information about DUMP via Task Manager.
2.03	22/12/2014	Review information about DUMP via Task Manager for 32bit.
2.04	19/03/2015	Review information about TSP DUMP for 32 and 64 bit OS.
2.05	19/08/2015	Added information about Starting traces automatically (via Diagnosis tool).
2.06	21/08/2015	Added information about detailed Audio logs.
2.07	25/09/2015	Review information about WinDbg DUMP and url.
2.08	08/10/2016	Added information about Device audio Log (Wav log) and contend review of other chapters.
2.09	04/03/2016	Review and included new command about TSP chapter 4.10.
2.10	20/05/2016	Review infos about Debug tools(Annex B and C) and changed Unify address. Also added details about how to fix TSP issues preventing HFA to work ().
2.11	02/01/2017	Added tool to automatically collect infos and print screens of test steps performed. Chapter 4.17 about PSR.
2.12	18/01/2019	Changed new details about audio logs. Chapter 4.15 & 4.16.
2.13	19/03/2019	Changed new details about audio logs. Chapter 4.15 (HFA).
2.14	30/04/2021	Changed M5T details. Chapter 4.11.
2.15	05/10/2021	Added more info about M5T details. Chapter 4.11.
2.16	27/01/2023	General Review of complete trace. General Specific Updated chapters 4 and review/ created chapter 5 (DUMPs).

2 About this guide

Generally, in case of issues related to clients (e.g. the clients stop working), traces must be collected from customers to allow developers identify and resolve errors.

The sooner the developers get the correct and precise traces, the better the chances to find out the possible sources of errors to any given customer problems.

It is important to allow technicians and sellers support to get the correct/precise traces at the first visit at customer's site.

This guide provides useful information and hints about how to install correctly the OpenScape Desktop Client (ODC) and get/extract the correct trace/information depending of the provider used by the client (**SIP** or **HFA Stackless**).

3 Traces required

3.1 Using ODC with SIP Provider

If ODC is configured as SIP Provider, the following traces are required:

1) Open **OptiClient GUI** or **OpenScape Desktop Client GUI**.

> **NOTICE:** It is important to start all traces/logs (see **2.4**) **BEFORE** starting to use **ODC-PE**.

2) Locate the **MSI** trace file.

3) Locate the **2 Installation** logs (generally stored in the %TEMP% folder).

The names of the logs include the version number.

Example:

For ODC-PE **V7 R1.47.93**, the 2 installation log files are:

- Unify OpenScape Client_**70.1.47.0093**_MSI.txt
- Unify OpenScape Client_**70.1.47.0093**_Setup.txt

4) Locate the **ODC-PE Configuration** files.

5) Analyse the **network traffic** on the PC where ODC is installed. You can use **Wireshark** or any other network traffic analyzer.

This trace is needed in case of issues with payload/calls.

6) Take a note of the **TIME** when the **error** occurred (format **hour: minutes**).

7) Take a note of the **LOGIN** and **PROFILE** used when the **error** occurred.

8) A potential error cause is high use of PC processor. Therefore, make sure to take a note of the following:

- Windows version used: e.g. Windows 10/11 (SP, last KB installed)
- The amount (%) of **CPU** used when the issue occurred.
- The amount of **RAM** memory used when the issue occurred.
- The amount of **FREE RAM** memory during ODC usage.

9) Describe how to reproduce the error in a **step-by-step** approach.

Example:

PartyA_Nr8900 calls PartyB_Nr8901 at 15:23hrs.

PartyB_Nr8901 hears the ring at 15:23hrs.

PartyB_Nr8901 accepts the ring at 15:24hrs.

… etc

Error: after X minutes/seconds the call is dropped at 15:26hrs.

10) Include screen shots or videos of the error if it is difficult to explain the details.

It is recommended to collect screen shots or videos at full screen to better present the steps, error and the time stamp of error.

11) Once all traces are collected, deactivate the traces.

3.2 Using ODC with HFA Provider (or HFA Stackless)

If ODC is configured as HFA Provider, the following traces are required:

1) Open **OptiClient GUI** or **OpenScape Desktop Client GUI**.

> **NOTICE:** It is important to start all traces/logs (see **2.4**) **BEFORE** starting to use **ODC-PE**.

2) Locate the **SCI** trace file.

3) Locate the **SoftOla** trace file.

4) Locate the **2 Installation** logs (generally stored in the %TEMP% folder).

The names of the logs include the version number.

Example:

For ODC-PE **V7 R1.47.83**, the 2 installation log files are:

- Unify OpenScape Client_**70.1.47.0083**_MSI.txt
- Unify OpenScape Client_**70.1.47.0083**_Setup.txt

5) Locate the **ODC-PE Configuration** files.

6) Analyse the **network traffic** on the PC where ODC is installed. You can use **Wireshark** or any other network traffic analyzer.

This trace is needed in case of issues with payload/calls.

7) Take a note of the **TIME** when the **error** occurred (format **hour: minutes**).

8) Take a note of the **LOGIN** and **PROFILE** used when the **error** occurred.

9) A potential error cause is high use of PC processor. Therefore, make sure to take a note of the following:

- Windows version used: e.g. Windows 10/11 (SP, last KB installed)
- The amount (%) of **CPU** used when the issue occurred.
- The amount of **RAM** memory used when the issue occurred.
- The amount of **FREE RAM** memory during ODC usage.

10) Describe how to reproduce the error in a **step-by-step** approach.

Example:

PartyA_Nr8900 calls PartyB_Nr8901 at 15:23hrs.

PartyB_Nr8901 hears the ring at 15:23hrs.

PartyB_Nr8901 accepts the ring at 15:24hrs.

… etc

Error: after X minutes/seconds the call is dropped at 15:26hrs.

11) Include screen shots or videos of the error if it is difficult to explain the details.

It is recommended to collect screen shots or videos at full screen to better present the steps, error and the time stamp of error.

12) Once all traces are collected, deactivate the traces.

3.3 Using ODC-WE (only for legacy)

If ODC-WE is used, the following traces are needed:

- The traces described in Using ODC with SIP Provider on page 6 and Using ODC with HFA Provider (or HFA Stackless) on page 7, depending on the provider type.
- **WebClient local** traces (see **2.4**).
- **WebClient server** traces.

For more information about **WebClient local** and traces **WebClient server** traces, see:

http://gvs-ftp.cycos.com/tiki-index.php?page=gvs_tiki_know_Webclient_core

For more detailed WebClient traces, including **DEBUG** and **TRACE** messages, the value of the **LOG_INIT_LEVEL4J** and **LOG_INIT_LEVEL** variables need to be set to **5** in the **global.cfg** file:

```
/opt/siemens/HiPathCA/config/common/global.cfg
# Log size depending on 8k scenario
<?x if(<?x $IS_HP8K ?> == "1") ?>
<?x setvar (LOG_INIT_LEVEL4J = "5" ) ?>
<?x setvar (LOG_INIT_LEVEL = "5" ) ?>
<?x setvar (LOG_INIT_MAXDAYS = "3" ) ?>
<?x setvar (LOG_INIT_MAXFILESPERDAY = "15" ) ?>
<?x else ?><?x setvar (LOG_INIT_LEVEL4J = "5" ) ?>
<?x setvar (LOG_INIT_LEVEL = "5" ) ?>
<?x setvar (LOG_INIT_MAXDAYS = "5" ) ?>
<?x setvar (LOG_INIT_MAXFILESPERDAY = "30" ) ?>
<?x endif ?>
```

To apply these changes, you need to **restart** the **WebClient** components (WebClient_FE & WebClient_BE).

WebClient traces that show the most important information are marked as **P[sS][mM]***.

The most important files to be reviewed within development are located in the following path `/var/siemens/common/log/webclient/<date>/P[sS][mM]*`:

- `PsmApplicationService-*.log`
- `PSMService-*.log`

In /opt/siemens/HiPathCA/config/services/default/PSMService/PSM.cfg you might also want to activate CSTA tracing by activating, if there are call flow errors.

In case of call flow errors, you can also activate CSTA tracing using the following path `/opt/siemens/HiPathCA/config/services/default/PSMService/PSM.cfg`.

Example:

```
CSTA.traceSubDir = csta
```

NOTICE:

It is **recommended** to attach the entire **WebClient log** folder.

Client-side traces

For issues related to non-responsive WebClient, additional **browser side traces** need to be provided. For this, set `:trace=on` in the `"< Name or Number >"` input file of WebClient UI.

Once the **trace** parameter is set to on, a pop-up window with the trace log is displayed. When an issue occurs, copy the content of the trace log and paste it into a text file. After capturing the trace log, close the pop-up window and set `:trace=off` in the `"< Name or Number >"` input file of WebClient UI.

3.4 Activating Client Traces via Diagnosis Tool

You can use the **OpenScapeClientDiagnosis.exe** tool to activate or deactivate specific traces from Clients (e.g. ODC-PE or ODC-WE), such as:

- OC traces
- MSI traces
- SCI traces
- SoftOla traces

You can securely collect all Client traces and additional system details via a button displayed in the tool.

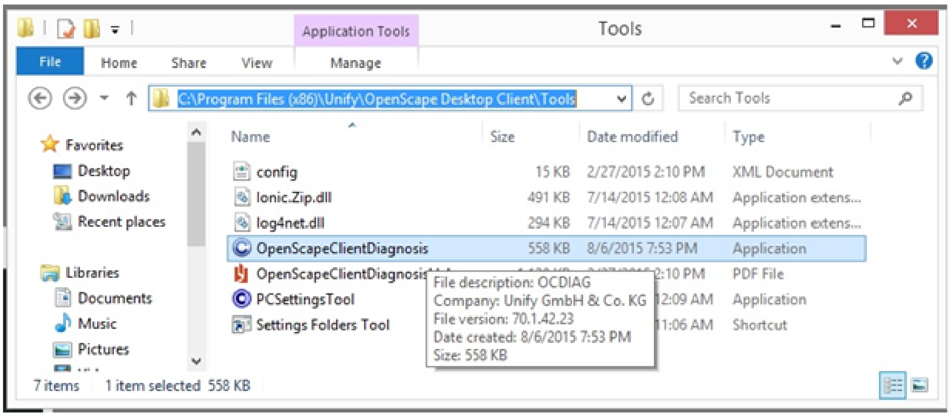
The Diagnosis tool is installed within the same base folder as the ODC-PE / WE, inside **Tools** subfolder.

3.4.1 Starting Diagnosis Tool

The Diagnosis tool is located in the following path:

```
C:\Program Files (x86)\Unify\OpenScape Desktop Client
\Tools:
```

To start the Diagnosis tool, run the **OpenScapeClientDiagnosis.exe** file.



C:\Program Files (x86)\Unify\OpenScape Desktop Client\Tools

Name	Size	Date modified	Type
config	15 KB	2/27/2015 2:10 PM	XML Document
Ionic.Zip.dll	491 KB	7/14/2015 12:08 AM	Application extens...
log4net.dll	294 KB	7/14/2015 12:07 AM	Application extens...
OpenScapeClientDiagnosis	558 KB	8/6/2015 7:53 PM	Application
OpenScapeClientDiagnosis		2:10 PM	PDF File
PCSettingsTool		2:09 AM	Application
Settings Folders Tool		1:06 AM	Shortcut

File description: OCDIAG
Company: Unify GmbH & Co. KG
File version: 70.1.42.23
Date created: 8/6/2015 7:53 PM
Size: 558 KB

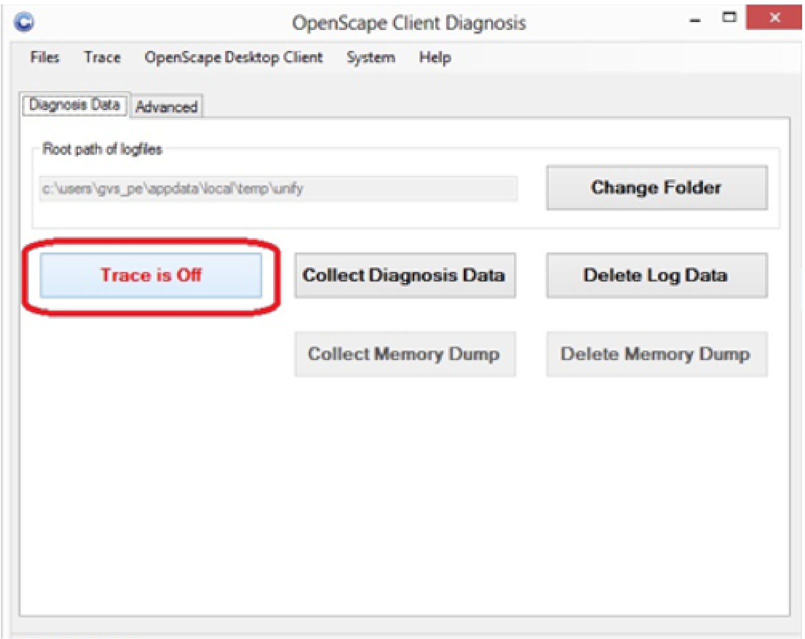
Figure 1: Starting the Diagnosis tool

NOTICE: If your operating system is Windows 32bit, the Diagnosis tool will be located in the following path:

```
C:\Program Files\Unify\OpenScape Desktop Client
\Tools
```

3.4.2 Configuring Traces

By default, the Traces are stopped on the Diagnosis tool.



Root path of logfiles: c:\users\gvs_pe\appdata\local\temp\unify

Figure 2: Configuring traces

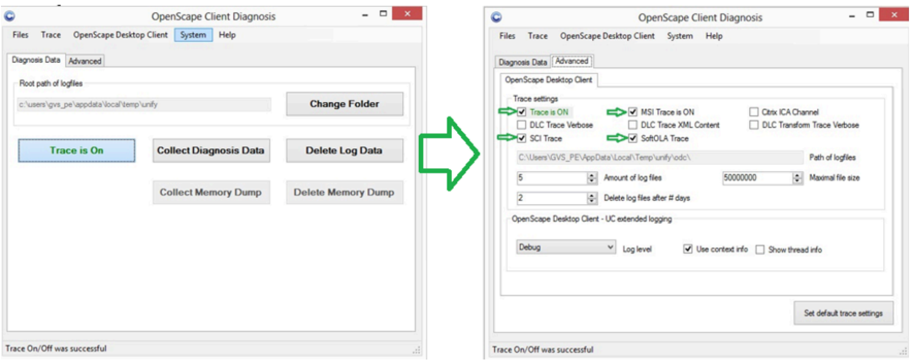
To start or stop collecting traces with the Diagnosis tool, click the **Trace** button:

- When the button displays **Trace is Off**, traces are stopped.
- When the button displays **Trace is On**, traces are started.

You can configure the traces you want to collect, using the following steps:

1) Navigating to the **Advanced** tab.

2) Select the traces you want to collect in the **Trace settings** area.



Root path of logfiles: c:\users\gvs_pe\appdata\local\temp\unify

Trace On/Off was successful

Trace settings

☑ Trace is ON ☑ MSI Trace is ON □ Citrix ICA Channel
□ DLC Trace Verbose □ DLC Trace XML Content □ DLC Transform Trace Verbose
☑ SCI Trace ☑ SoftOLA Trace

C:\Users\GVS_PE\AppData\Local\Temp\unify\odc\ Path of logfiles

5 Amount of log files 50000000 Maximal file size

2 Delete log files after # days

OpenScape Desktop Client - UC extended logging

Debug Log level ☑ Use context info □ Show thread info

Trace On/Off was successful

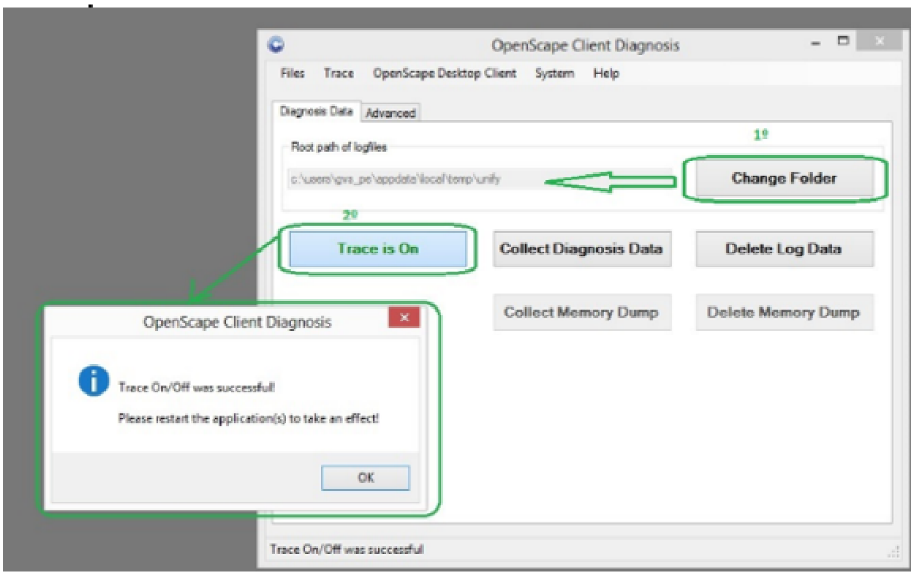
Figure 3: Configuring the traces

You can change the path where traces are stored, using the following steps:

1) Click **Change Folder** and select the path where you want to store traces.

2) Click **Trace is On** to start collecting traces.

3) Restart the tool for changes to take effect.



Root path of logfiles: c:\users\gvs_pe\appdata\local\temp\unify

Trace On/Off was successfull!

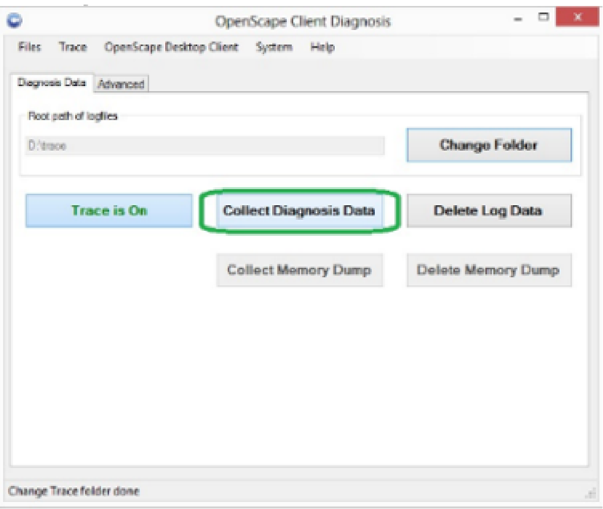
Please restart the application(s) to take an effect!

Trace On/Off was successful

Figure 4: Changing the traces folder

3.4.3 Collecting Traces

When an error occurs, close the client and open the Diagnosis tool. Click **Collect Diagnosis Data** to start collecting trace logs.

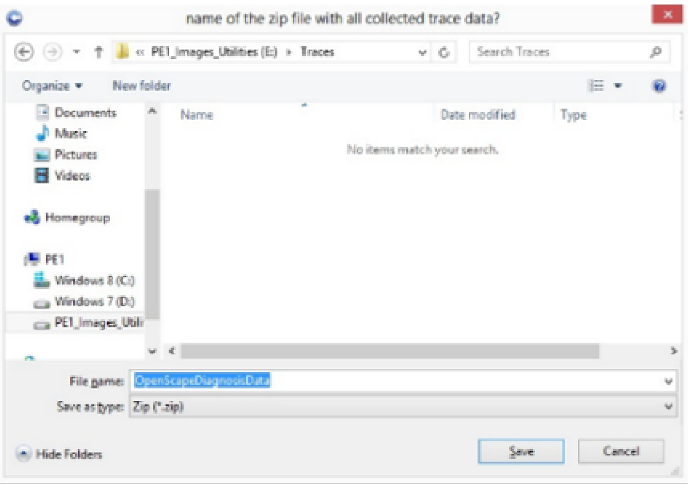


Root path of logfiles: D:\trace

Change Trace folder done

Figure 5: Collecting all client traces

A now window displays, asking you to select a location to store the traces. The traces will be available in the selected path as a ZIP file.



name of the zip file with all collected trace data?

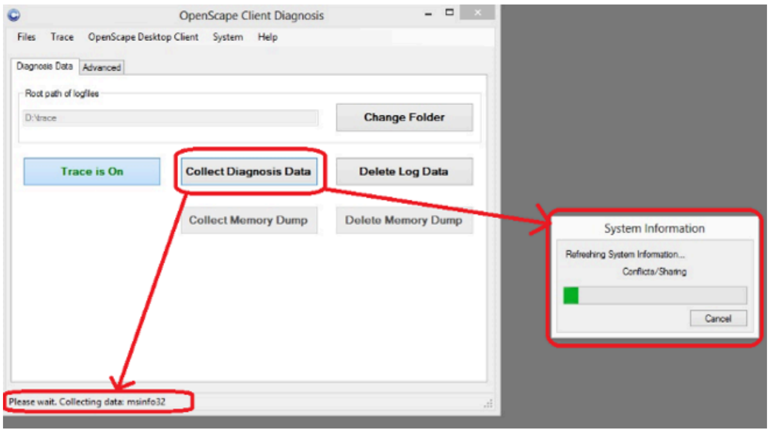
No items match your search.

File name: OpenScapeDiagnosisData

Save as type: Zip (*.zip)

Figure 6: Configuring the traces

NOTICE: It might take a few minutes until all traces are collected and the ZIP file is generated.



OpenScape Client Diagnosis

Root path of logfiles: D:\trace

System Information

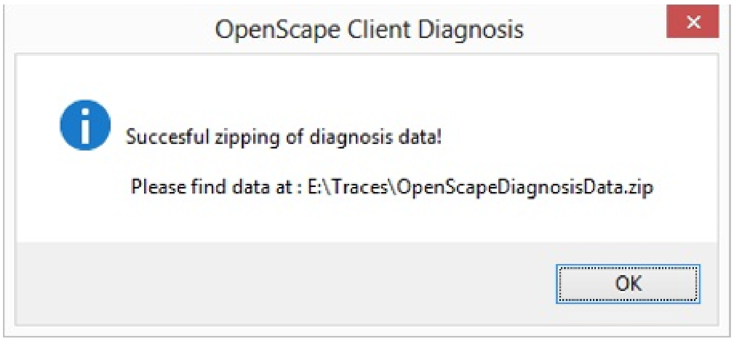
Refreshing System Information...

Conflicts/Sharing

Please wait. Collecting data: msinfo32

Figure 7: Diagnosis Tool generating the ZIP file with traces

After all traces are collected and the ZIP file is generated, a message is displayed in the Diagnosis tool.



OpenScape Client Diagnosis

Succesful zipping of diagnosis data!

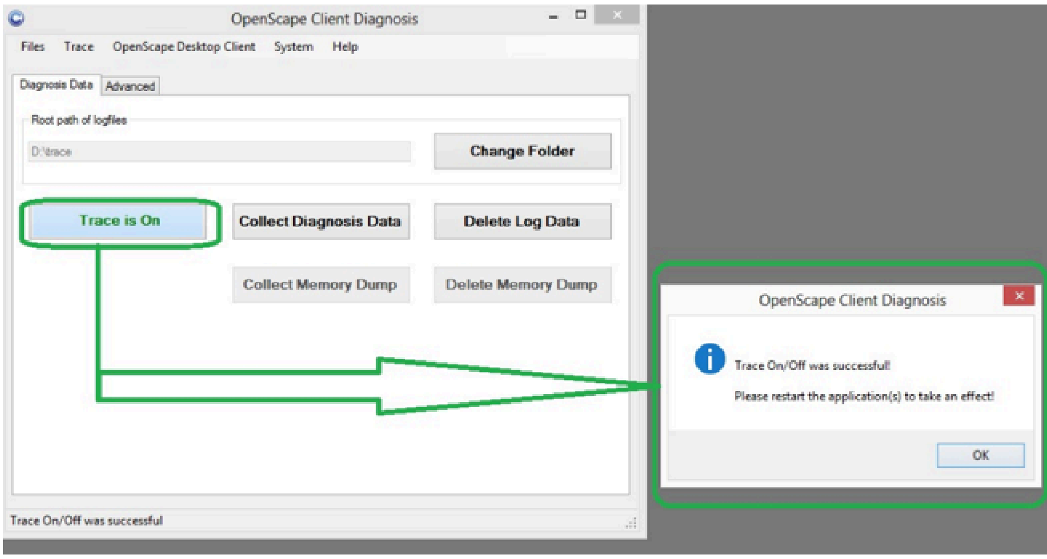
Please find data at : E:\Traces\OpenScapeDiagnosisData.zip

Figure 8: Successful message for collecting and zipping traces

3.4.4 Disabling Traces

It is recommended to disable the traces from the Diagnosis tool, after you finish collecting them.

To disable traces, click **Trace is On** and restart the Diagnosis tool for the changes to take effect.



Trace On/Off was successful!

Please restart the application(s) to take an effect!

Trace On/Off was successful

Figure 9: Disabling traces

3.5 Regedit file Settings

Generally, all OpenScape Desktop Client traces (**OC, MSI, SCI** and **SoftOla**) must be activated or deactivates via the Diagnosis tool. For more information, see Activating Client Traces via Diagnosis Tool on page 9.

There are specific cases when traces must to activated or deactivated manually, via a **regedit file**.

Regedit files provide a secure and consistent way to configure trace settings. Using regedit files, you can **collect all necessary traces** and **speed up the (de)activation of traces**.

Examples of using regedit files:

Traces for SIP Provider

64 bits

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Siemens\OpenScape
\Trace]
"WriteToFile"=dword:00000001
"MSITrace"=dword:00000001
"PathFile"="c:\\traces\\"
"MSItraceFile"="c:\\traces\\msi.trc"
```

32 bits

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Siemens\OpenScape\Trace]
```

```
"WriteToFile"=dword:00000001
"MSITrace"=dword:00000001
"PathFile"="c:\\traces\\"
"MSItraceFile"="c:\\traces\\msi.trc"
```

Traces for HFA Stackless Provider

64 bits

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Siemens\OpenScape
\Trace]
"WriteToFile"=dword:00000001
"PathFile"="c:\\traces\\"
[HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Siemens\SoftOLA
\Trace]
"WriteToFile"=dword:00000001
"PathFile"="c:\\traces\\"
"SciTrace"=dword:00000001
"SciPathFile"="c:\\traces\\sci.trc"
```

32 bits

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Siemens\OpenScape\Trace]
"WriteToFile"=dword:00000001
"PathFile"="c:\\traces\\"
[HKEY_LOCAL_MACHINE\SOFTWARE\Siemens\SoftOla\Trace]
"SciTrace"=dword:00000001
"WriteToFile"=dword:00000001
"PathFile"="c:\\traces\\"
"SciPathFile"="c:\\traces\\sci.trc"
```

3.6 Quick Review and Planning

When reviewing traces, the technicians and support teams analyze the following:

- The client used by the customer (ODC-PE version)
- The Provider type used (SIP or HFA Stackless)
- In crash or freeze scenarios, the **DUMP 32 bit** trace must be provided.

  You can use one of the following tools to collect **DUMP 32 bit** logs:

  – Windows **TaskManager 32 bit**
  – **ADPlus**
  – **Windbg**, when it is not possible to use **TaskManager 32 bit** or **ADPlus**

    E.g. if the software does not crash but it freezes or when it crashes during the start of the client.
- For issues related to communication or networking, you must also provide WireShark traces.
- The **detailed** description of the issue, **with a step by step approach** and the **timestamp** of the error.

  This allows **GVS** or **Development** teams to reproduce the error in the lab.
- **Print screens** and/or **Video** (in **FULLSCREEN**) to provide a better overview of the issue.

After analysing information provided, further details or traces might be requested by the support teams.

3.7 Summary of traces

For new Customer Tickets (PRB), the following traces must be provided:

Traces Names / types	ODC-**PE** **HFA Stackless**	ODC-**PE** **SIP**	ODC-PE HFA H323 *
GUI / Pearl Trace (**OC**….**trc**)	Yes	Yes	Yes
Softola Trace	Yes		Yes
SCI trace	Yes		Yes
MSI trace		Yes	
DUMP 32 bit (if crash/ freeze)	Yes	Yes	Yes
Audio Device (**WAV** log)	D	D	D
MS PerfMon log	D	D	D
Installation log	Yes	Yes	Yes
Configuration files	Yes	Yes	Yes
Wireshark Trace	Yes	Yes	Yes
TSP Trace	D		D
M5T trace		D	
RV Trace	D		D

Yes: mandatory to be attached

D: only requested on specific development requests

*: HFA H323 it is **NOT** supported anymore.

3.8 Trace description

This chapter describes how to activate different traces and trace levels for OpenScape Desktop Client. You can create or activate the following traces, depending on the errors found:

- **optiClient GUI** or **OpenScape Desktop Client GUI trace** contains:

  The trace of the user **interface / GUI / Pearl**.

  The trace of the **SIP Engine**.
- **Softola trace** contains:

  The trace of the communication between OpenScape Desktop Client and the communication system, the configuration and control of the SCI.

- **SCI trace** contains:

  The trace of the negotiation of the codecs.

  The control of the media stream between HiPath **HFA**and the soundcard.
- **MSI trace** contains:

  The trace of the negotiation of the codecs

  The control of the media stream between HiPath SIP and the soundcard.
- **RAD vision trace** contains:

  The trace of the **H323**connection.
- **M5T trace** contains:

  Traces of the M5T component (responsible for the SIP engine, at a lower level).
- **TSP trace** contains:

  Traces from communication features call control between SoftOLA and the PBX system.
- **WebClient trace** contains:

  Local traces of the webclient.
- **DLS traces** contains:

  Traces of the communication between ODC and DLS.
- **DUMP 32 bit** contains:

  DUMP traces in **32 bit** format, collected during the ODC crash.

  This type of trace can be collected via on of the following tools: Windows **TaskManager** (**32 bit** variation), **WinDbg**, **ADPlus**.
- Advanced **Audio logs**
- **Audio Device log (Mic wave)**
- **MS Performance Monitor**
- **PSR tool**
- **Wireshark** traces

  Specific log for problems related to networking, lost of packages, VPN, server issues (Citrix, etc). Traces should be collected via Wiresbhark or a similar tool.

3.9 Activating Traces/DUMPs

3.9.1 Activating OC/GUI – Trace

To activate the **OC/GUI-Trace**, you must create the following Registry – Key:

- For a **32-bit** system:

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Siemens\OpenScape\Trace]
```

- For a **64-bit** system:

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432node\Siemens\OpenScape\Trace]
```

Advanced options

- **Write trace to file**

  Create the following registry value in the **Trace** key:

  ```
  "WriteToFile"
  DWORD 1 = On, 0 = Off
  ```

- **Pathfile**

  Create the following registry value in the **Trace** key:

  ```
  "PathFile"
  STRING example: "C:\TEMP\"
  ```

  When you start a trace, a new file is created in the **C:\TEMP** folder. If you write a full path with a file name (e.g. " C:\TEMP\Tracefile.trc), all traces will be appended in the given file.

  – **PS1:** If the value **PathFile** is missing, trace files will be created under "Own Files" every time a trace is started, with a new file name.

  – **PS2:** The path used **must exist** on the system.

- **Free space**

  Create the following registry value in the **Trace** key:

  ```
  "FreeDriveSpacePerCent"
  DWORD 0-99
  ```

  Tracing is stopped if the there is no free space on the drive.

- **Last date**

  Create the following registry value in the **Trace** key:

  ```
  "LastTraceDateTime"
  STRING "200112201350"
  ```

  Tracing is stopped at 20.12.2001 13:50.

- **Max File Size**

  Create the following registry value in the **Trace** key:

  ```
  "MaxFileSize"
  DWORD x Bytes
  ```

  Tracing is stopped if the trace file has reached the given size (x Bytes).

- **Files**

  Create the following registry value in the **Trace** key:

  ```
  "Files"
  ```

```
DWORD 0-99
```

The tracing uses a circular buffer. This registry value is in correlation with **MaxFileSize**. If the value defined for the number of files is greater than 0, "~xx" will be added to the name of the trace files.

If a trace file reaches the size defined by **MaxFileSize**, a new file is created (with the next number) until the number defined by "Files" is reached. Then, the tracing will be written in the first file "~01".

If the **PathFile** value is similar to `C:\TEMP\`, then every trace that starts will create a new packet of files.

- **Delete** older trace files, when using the **Files** key

  Create the following registry value in the **Trace** key:

  ```
  "DeleteXDaysOldFiles"
  DWORD xx Days
  ```

  If you use the registry values **Files** and **PathFile** with a path similar to `C:\TEMP\`, then **every** trace that starts will create a new packet of files.

  To delete the older file packages on trace start, you can use this registry value. For this, enter the desired value:

  e.g. To delete all file packages older than one week (7 days), use value **7**.

3.9.2 Activating Softola – Trace

To activate the **Softola-Trace**, you must create the following Registry – Key:

- For a **32-bit** system:

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Siemens\SoftOla\Trace]
```

- For a **64-bit** system:

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432node\Siemens\SoftOla\Trace]
```

Advanced options

- **Tracelevel**

  Create the following registry value in the **Trace** key:

  ```
  "TraceLevel"
  DWORD 9
  ```

- **Write trace to file**

  Create the following registry value in the **Trace** key:

  ```
  "WriteToFile"
  DWORD 1 = On, 0 = Off
  ```

- **Pathfile**

  Create the following registry value in the **Trace** key:

  ```
  "PathFile"
  ```

```
STRING example: "C:\TEMP\"
```

When you start a trace, a new file is created in the **C:\TEMP** folder. If you write a full path with a file name (e.g. " C:\TEMP\Tracefile.trc), all traces will be appended in the given file.

- **PS1:** If the value **PathFile** is missing, trace files will be created under "Own Files" every time a trace is started, with a new file name.
- **PS2:** The path used **must exist** on the system.

- **Free space**

  Create the following registry value in the **Trace** key:

  ```
  "FreeDriveSpacePerCent"
  DWORD 0-99
  ```

  Tracing is stopped if the there is no free space on the drive.

- **Last date**

  Create the following registry value in the **Trace** key:

  ```
  "LastTraceDateTime"
  STRING "200112201350"
  ```

  Tracing is stopped at 20.12.2001 13:50.

- **Max File Size**

  Create the following registry value in the **Trace** key:

  ```
  "MaxFileSize"
  DWORD x Bytes
  ```

  Tracing is stopped if the trace file has reached the given size (x Bytes).

- **Files**

  Create the following registry value in the **Trace** key:

  ```
  "Files"
  DWORD 0-99
  ```

  The tracing uses a circular buffer. This registry value is in correlation with **MaxFileSize**. If the value defined for the number of files is greater than 0, "~xx" will be added to the name of the trace files.

  If a trace file reaches the size defined by **MaxFileSize**, a new file is created (with the next number) until the number defined by "Files" is reached. Then, the tracing will be written in the first file "~01".

  If the **PathFile** value is similar to `C:\TEMP\`, then every trace that starts will create a new packet of files.

- **Delete** older trace files, when using the **Files** key

  Create the following registry value in the **Trace** key:

  ```
  "DeleteXDaysOldFiles"
  ```

```
DWORD xx Days
```

If you use the registry values **Files** and **PathFile** with a path similar to `C:\TEMP\`, then **every** trace that starts will create a new packet of files.

To delete the older file packages on trace start, you can use this registry value. For this, enter the desired value:

e.g. To delete all file packages older than one week (7 days), use value **7**.

3.9.3 Activating SCI – Trace

To activate the **SCI-Trace**, you must set/create the following Registry – Value:

- For a **32-bit** system:

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Siemens\SoftOla\Trace]
```

- For a **64-bit** system:

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432node\Siemens\SoftOla\Trace]
```

This will create a trace file in the temp folder with the name **sciyyyymmddhhmmss.trace.log**, where **yyyymmddhhmmss** is the date an time when the tracing starts.

The location of the temp folder depends on the setup of the TEMP environment variable. Typically, the temp folder is located at: `%temp%`.

The tracing stops automatically when the **drive space is less than 5%**.

When the trace file is created and there are more than **10 old tracefiles** in the temp folder, the **oldestfile** will be **deleted**.

3.9.4 Activating MSI – Trace

To activate the **MSI-Trace**, you must set/create the following Registry – Value:

- For a **32-bit** system:

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Siemens\OpenScape\Trace]
```

- For a **64-bit** system:

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432node\Siemens\OpenScape\Trace]
```

This will create a trace file in the temp folder with the name **MSITraceyyyymmddhhmmss.txt**, where **yyyymmddhhmmss** is the date an time when the tracing starts.

The location of the temp folder depends on the setup of the TEMP environment variable. Typically, the temp folder is located at: `%temp%`.

It is possible to configure the path and file name of the trace file via the following registry key:

```
“MSITraceFile” STRING
```

- For a **32-bit** system:

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Siemens\OpenScape\Trace]
```

- For a **64-bit** system:

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432node\Siemens\OpenScape\Trace]
```

This parameter must contain a valid path and file name, similar to `C:\temp\msi.txt`.

If the path and file name are configured, the date and time will be added to the file name: e.g. `C:\temp\msiyyyymmddhhmmss.txt`.

When the trace file is created and there are more than **10 old tracefiles** in the temp folder, the **oldestfile** will be **deleted**.

3.9.5 OpenScape Desktop Client - Installation log files

During the installation of the client, the following log files are created in the following `%temp%` folder:

```
Siemens OpenScape Desktop Client_”versionnumber”_MSI.txt
Siemens OpenScape Desktop Client_”versionnumber”_Setup.txt
```

3.9.6 OpenScape Desktop Client - Installation log files

The client configuration files are stored in XML format.

NOTICE: All files in the configuration files must be archived and attached to the corresponding ticket.

The default folder with the OpenScape Desktop Client configuration is located in the following path:

`%appdata%\Unify\OpenScape` (or `%appdata%\Siemens\OpenScape` for older versions).

If the path has changed, you can use the **Unify.OptiClient.SettingsFoldersTool.exe** tool to identify the new location. This tool available under:

- For a **32-bit** system:
`%programfiles%\Unify\OpenScape Desktop Client\Client`
- For a **64-bit** system:
`%programfiles(x86)%\Unify\OpenScape Desktop Client\Client`

If the path did not change, the fields displayed in the tool are empty.

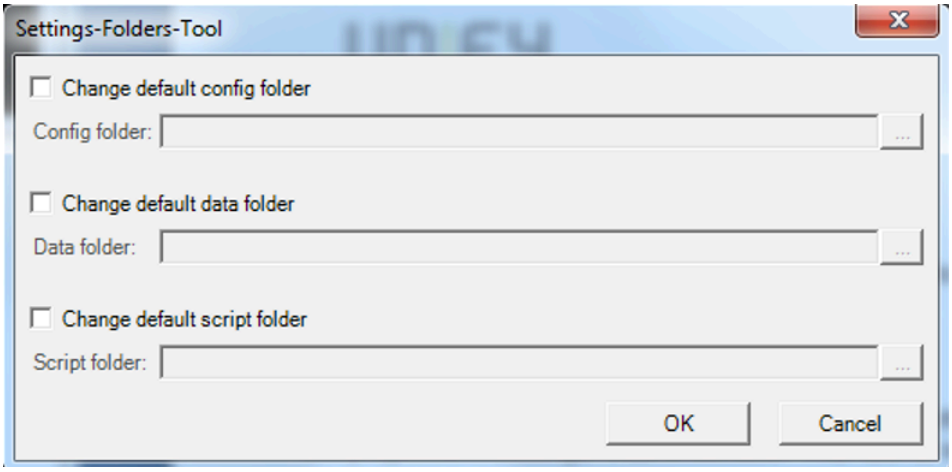

Settings-Folders-Tool
☐ Change default config folder
Config folder:
...
☐ Change default data folder
Data folder:
...
☐ Change default script folder
Script folder:
...
OK
Cancel

Figure 10: Viewing the configuration path

3.9.7 TSP Trace

In case of TSP Trace, you need to first determine the TSP version running on your computer.

Use the following command to determine to identify the TSP version:

```
C:\WINDOWS\system32>wmic product
where"Name like '%Unify TSP%'" get Name, Version,
 IdentifyingNumber
```

Example results:

IdentifyingNumber: {C3C1BF22-F499-4DB3-874B-EB792EE63C9E}

Name: Unify TSP x64

Version: 1.0.0

To collect TSP traces, follow the steps below:

1) Download or install the Microsoft tools used for collecting traces (Dbgview or Windbg).
2) Activate the traces via regedit (based on the PBX used, regardless of the OS).
3) Collect the traces after the tests are performed

3.9.7.1 Activating the trace

To collect TSP traces (as 4.10.2), you need to activate the TSP trace first.

You must create the traces for OpenScape Business (OSBiz) and OpenScape 4000 (OS4K) separately:

- For **OSBiz (**or **H3K**, **OSO**, or any other PBX **except for theHP4K family)**:

  Set/create the Registry – Value:
  `DBGTrace"= dword:00000001`

  in the registry key:

  ```
  [HKEY_LOCAL_MACHINE\SOFTWARE\Siemens\optiClientTSP Hicom
   150\Trace]
  ```

  If requested by GVS or Development teams, add the following registry value `DBGTraceEnh"= dword:00000001` in the same registry key.
- For **H4K**:

  Set/create the Registry – Value:
  `DBGTrace"=dword:00000001`

  in the registry key:

  ```
  [HKEY_LOCAL_MACHINE\SOFTWARE\Siemens\optiClientTSP
   HiPath 4000 V2\Trace]
  ```

  If requested by GVS or Development teams, add the following registry value `DBGTraceEnh"= dword:00000001` in the same registry key.

NOTICE: To activate the trace, you must restart the **telephony** service (or, if it is not possible, **restart the pc)** .

NOTICE: Such activation must be done for all OS and all types of PBX.

3.9.7.2 Collecting the trace

Regardless of OS and PBX, it is necessary to use a compatible tool (e.g. Dbgview or Windbg) to collect the TSP trace activated before via regedit.

Windows XP

If the OS is Windows XP, you must first start the Dbgview tool to collect the trace, before starting the Client. Once the test is completed, save the file.

Windows 7 and Higher

If the OS is Windows 7 or higher, the process you need to follow to collect the trace is more complex. The **svchost** process which is the parent process of telephony service should be attached to the Windbg tool. To do so, follow the steps below:

1) Start WinDbg.
2) Attach the **svchost** process that has the telephony service under it.
3) In the command line of WinDbg, run the `sxe  *` command, then the `g` command.
4) Execute the test.
5) In WinDbg, break the running process.
6) Save the file from WinDbg.
7) Deattach the process from WinDbg.

For more information, please refer to Annex C.

3.9.7.3 Fixing TSP issues after install

Regardless of ODC-PE version installed, TSP issues might occur due to wrongly deleted regedit keys during the optimization of the regedit tool and/or the uninstall/setup of other programs.

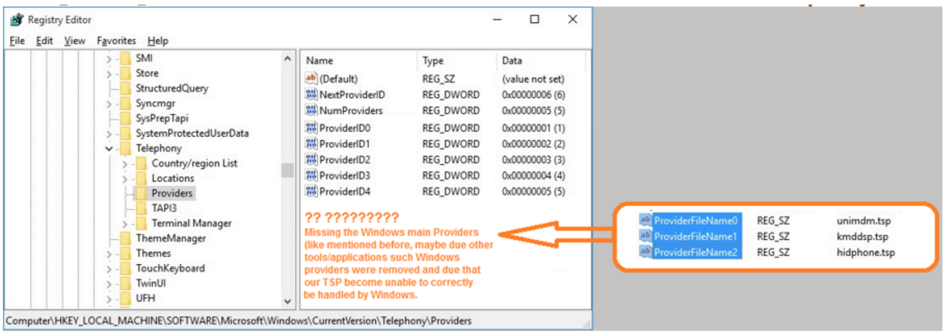
PE HFA displays **error** messages in case of **corrupted/incompleteTSP** installation.

Example:

- "**TAPI** Provider can **not** be found "
- “**HiPath-Provider 5.1** is **not** connected ”.

Example: error 1

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Telephony\Providers



Name	Type	Data
(Default)	REG_SZ	(value not set)
NextProviderID	REG_DWORD	0x00000006 (6)
NumProviders	REG_DWORD	0x00000005 (5)
ProviderID0	REG_DWORD	0x00000001 (1)
ProviderID1	REG_DWORD	0x00000002 (2)
ProviderID2	REG_DWORD	0x00000003 (3)
ProviderID3	REG_DWORD	0x00000004 (4)
ProviderID4	REG_DWORD	0x00000005 (5)

?? ?????????

Missing the Windows main Providers (like mentioned before, maybe due other tools/applications such Windows providers were removed and due that our TSP become unable to correctly be handled by Windows.

ProviderFileName0	REG_SZ	unimdm.tsp
ProviderFileName1	REG_SZ	kmddsp.tsp
ProviderFileName2	REG_SZ	hidphone.tsp

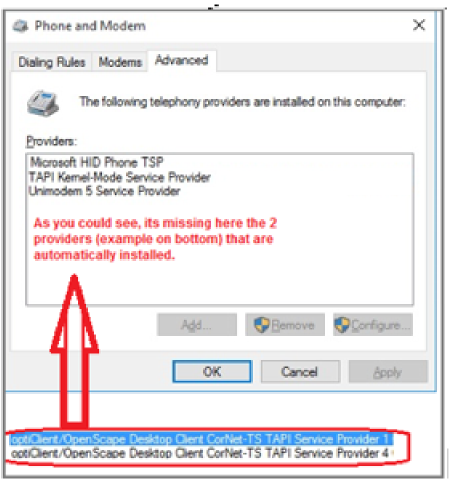
Computer\HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Telephony\Providers

Figure 11: Example of problematic regedit (missing the main 3 provider's details)

Example: error 2

Several issues can be identified via the **Phone & Modem** Windows module.

If the ODC-PE providers are missing, then the setup is not correct.



The following telephony providers are installed on this computer:

Providers:

Microsoft HID Phone TSP
TAPI Kernel-Mode Service Provider
Unimodem 5 Service Provider

As you could see, its missing here the 2 providers (example on bottom) that are automatically installed.

optiClient/OpenScape Desktop Client CorNet-TS TAPI Service Provider 1
optiClient/OpenScape Desktop Client CorNet-TS TAPI Service Provider 4

Figure 12: Example of problem confirmation via "Phone and Modem" module.

If the provider does **not exist** or it is **empty**, then you can only fix this issue manually via regedit.

First, you must check if the issues are similar to the previous ones. To check this, see above errors with Customer's **regedit keys** (if it **exists**) and if it has **contents**.

If the problem is the same as a previous one, follow the steps below (**administrator** privileges are **mandatory**):

1) Close ODC-PE.

2) Stop the **Telephony** service.

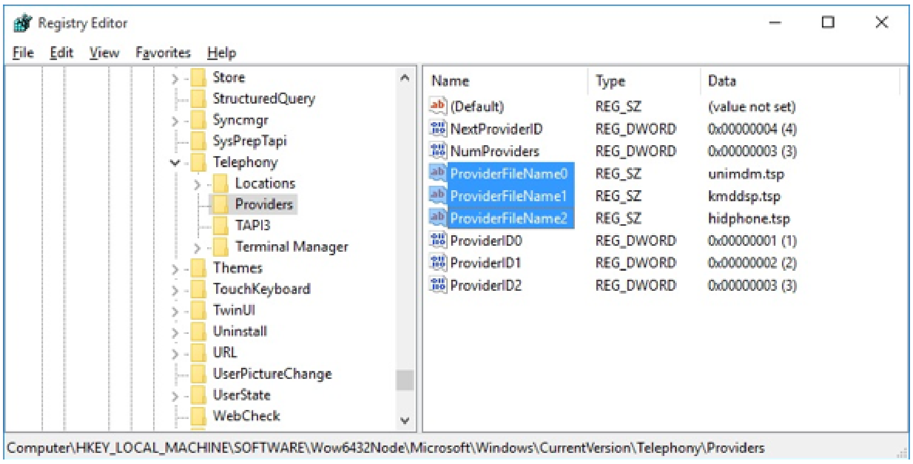
3) The **registry** key is **created**:

```
"HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Telephony\Providers"
```

4) The contents of the registry key:
`"HKEY_LOCAL_MACHINE\SOFTWARE\ Wow6432Node \Microsoft\Windows\CurrentVersion \Telephony\Providers"`

are copied to the registry key:
`"HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Telephony\Providers"`

Example of a normal/clean **Wow6432Node** regedit:



Name	Type	Data
(Default)	REG_SZ	(value not set)
NextProviderID	REG_DWORD	0x00000004 (4)
NumProviders	REG_DWORD	0x00000003 (3)
ProviderFileName0	REG_SZ	unimdm.tsp
ProviderFileName1	REG_SZ	kmddsp.tsp
ProviderFileName2	REG_SZ	hidphone.tsp
ProviderID0	REG_DWORD	0x00000001 (1)
ProviderID1	REG_DWORD	0x00000002 (2)
ProviderID2	REG_DWORD	0x00000003 (3)

Computer\HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Microsoft\Windows\CurrentVersion\Telephony\Providers

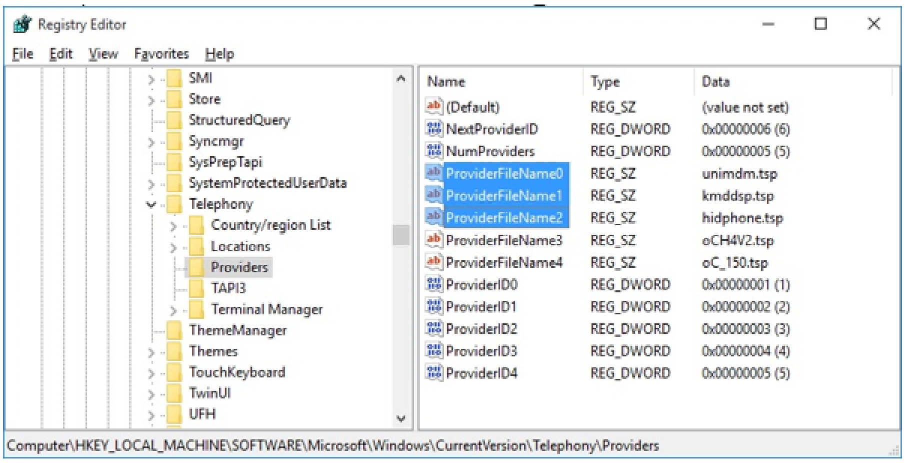
Figure 13: Example of normal/Clean Wow6432Node regedit (for be used to edit the 64-bit key) .

5) Start the **Telephony** service.

6) The "**TSPx64**" installation is **repaired**. If the repair does not add the ODC's TSP providers, the TSP must be uninstalled and then installed again.

7) Start ODC-PE.

Example after the issues with regedit are fixed:

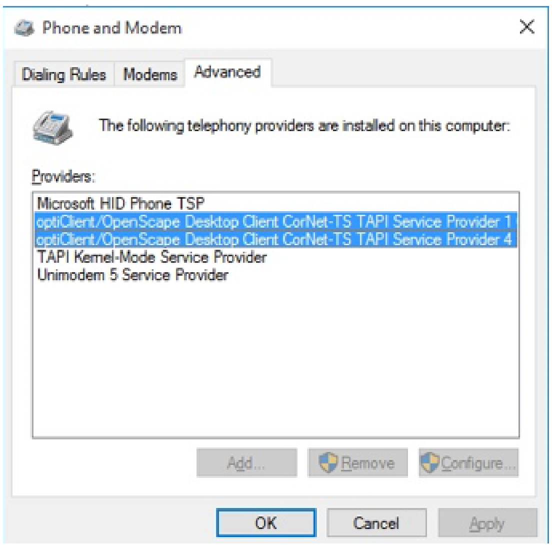


Name	Type	Data
(Default)	REG_SZ	(value not set)
NextProviderID	REG_DWORD	0x00000006 (6)
NumProviders	REG_DWORD	0x00000005 (5)
ProviderFileName0	REG_SZ	unimdm.tsp
ProviderFileName1	REG_SZ	kmddsp.tsp
ProviderFileName2	REG_SZ	hidphone.tsp
ProviderFileName3	REG_SZ	oCH4V2.tsp
ProviderFileName4	REG_SZ	oC_150.tsp
ProviderID0	REG_DWORD	0x00000001 (1)
ProviderID1	REG_DWORD	0x00000002 (2)
ProviderID2	REG_DWORD	0x00000003 (3)
ProviderID3	REG_DWORD	0x00000004 (4)
ProviderID4	REG_DWORD	0x00000005 (5)

Computer\HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Telephony\Providers

Figure 14: Example of a fixed 64-bit regedit key

Example after the issues with Windows **Phone and Modem** module are fixed:



The following telephony providers are installed on this computer:

Providers:

Microsoft HID Phone TSP
optiClient/OpenScape Desktop Client CorNet-TS TAPI Service Provider 1
optiClient/OpenScape Desktop Client CorNet-TS TAPI Service Provider 4
TAPI Kernel-Mode Service Provider
Unimodem 5 Service Provider

Figure 15: Example of a “Phone & Modem” module fixed

3.9.7.4 Activating M5T Trace

To **activate** the **M5T-Trace**, you must set/create the following Registry – Value `“m5tdebugtrace"=dword:00000001`

- For a **32-bit** system:

`[HKEY_LOCAL_MACHINE\SOFTWARE\Siemens\Sipengine]`

- For a **64-bit** system:

`[HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432node\Siemens\Sipengine]`

Additional details:

The **Sipengine** key will not be automatically created. It must be manually created by the user or admin:

1) With the “**Siemens**” key selected, right-click and select “**New => Key** ”.
2) Rename it to “**SipEngine**”.
3) Inside it, create the **DWORD“m5tdebugtrace"** with the value **00000001**.

Hints

If the system becomes very slow (that prevents or makes it difficult to perform the tests), collect the logs using the following method:

> **IMPORTANT:** This method is only recommended if the system/ PC becomes slow.

By default, once the key with value = 00000001 is created, the information will be logged in the Gui-Trace. However, it is possible to configure the path and file name of the trace file:

To set the path and file name, use the registry key **stacktracepath** (key type: REG_SZ or String).

Example:

stacktracepath=c:\traces\M5Ttrace.txt

- For a **32-bit** system:

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Siemens\Sipengine]
```

- For a **64-bit** system:

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432node\Siemens\Sipengine]
```

The parameter must be a valid path and file name, as for example: `c:\traces\M5Ttrace.txt`

> **NOTICE:**
>
> The file (mentioned via **stacktracepath**) will be **overwriting** every time **ODC** starts.

> **NOTICE:**
>
> When the **stacktracepath** registry key is created, the **M5T** logs will be stored in the following two locations: the **GUI-Trace** and the file specified by the **stacktracepath** key.

Therefore, for systems that become very slow, set a valid path to the **stacktracepath** key and the value **0x0** for the **m5tdebugtrace** key.

With such configuration, the **M5T** information will be stored **only** in the file specified by **stacktracepath** (and **not** in the **GUI-Trace**, making the system be less slow).

Disabling the M5T-Trace

To disable the trace, you must set the following Registry - Value:

```
“m5tdebugtrace"=dword:00000000
```

NOTICE: Setting this key to 0 will stop logging M5T within the OC/GUI log.

NOTICE: If **stacktracepath** configured, the log with still be collected (even with **m5tdebugtrace** = **0**).

To completely disable M5T logs:

- Set the value of **m5tdebugtrace** to **0** (or **deleted**)
- The **stacktracepath** key should be **deleted**.

⚠ **WARNING:** Activating the M5T trace will slow down the system. Only activate this trace if the GVS or development team requests it.

Additionally, disable the M5T trace as soon as the requested traces are collected.

3.9.8 Activating WebClient Traces

These traces will collect only information related to the interface.

To collect WebClient traces, run the following command in **WE**
```
:trace=on
```

The execution of this command will open a new window, where the interface information is logged.

After the test execution, copy this content of the new window, paste it into a text editor and save it.

Finally, run the following command to stop the traces:
```
:trace=off
```

NOTICE: The field is part of the main GUI. More precisely, the field is located where the user enters a phone number or a name to perform a search or DIAL.

3.9.9 Activating DLS Traces

It is recommended to activate these traces only in case of issues with the connection between ODC and DLS or if requested by the GVS or Development team.

To activate DLS traces, create the following keys:

```
DlcTraceVerbose=dword:00000001
DLCTransformTraceVerbose=dword:00000001
DLCTraceXMLContent=dword:00000001
```

Use the following registry keys:

- For a **32-bit** system:

`[HKEY_LOCAL_MACHINE\SOFTWARE\Siemens\OpenScape]`

- For a **64-bit** system:

`[HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432node\Siemens\OpenScape]`

3.9.10 Activating RV-Trace

The client includes the Radvision DLLs with the following compile option: **'NOLOGSUPPORT'= true**. Therefore, by default it is not possible to activate the RV-Trace.

Follow the steps below to activate and collect the RV trace:

1) Change the **original DLLs** from the **Client**, with the corresponding **DLLs** with LOG support.

   For DLLs log, navigate to the following path: `\Misc\Trace\RV`
2) Copy the **INI file**, located in the following path `\Misc\Trace\RV \rvtele.ini` and paste it into the Windows directory (`C:\WINDOWS`)
3) Restart the Client.
4) A new trace file is created in the temp folder with the following file name: name**rvtsp1.log**.

NOTICE: The location of the temp folder depends on the setup of the TEMP environment variable. Typically, the temp folder is located at: `%temp%`.

3.9.11 Advanced Audio Logs

If you experience audio issues, you must collect and provide Audio logs.

Audio Logs

To collect basic audio traces, you must activate the **AudioTrace** registry key. Currently, there is no support for tracing USB.

To activate audio traces, follow the steps below:

- **SIP Provider**

  64 bit: `HKLM\Software\WOW6432Node\Siemens\OpenScape`

  32 bit: `HKLM\Software\Siemens\OpenScape`

  Create a new **DWORD (32)** with “**AudioTrace**” set to **1**.

  The **WAVE Traces files** will be either stored in the same folder as the **GUI trace** or in the **%TEMP%** folder.

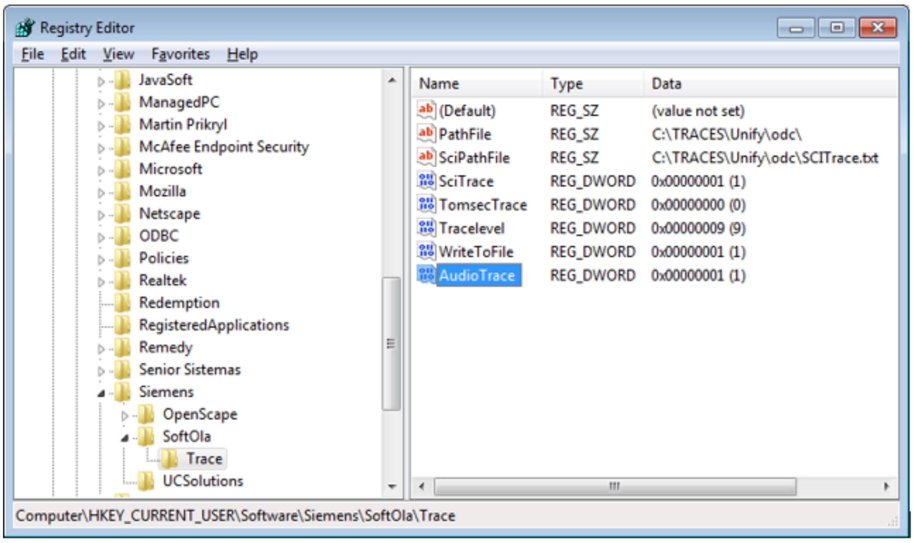
- **HFA Provider**

  Regedit: `HKCU\Software\Siemens\SoftOla\Trace`

  > **NOTICE:**
  >
  > The same folder is used for HFA, as for SIP.

  Create a new **DWORD (32)** with “**AudioTrace**” set to **1**.

  The **WAVE Traces files** will be either stored in the same folder as the **GUI trace** or in the **%TEMP%** folder.



JavaSoft
ManagedPC
Martin Prikryl
McAfee Endpoint Security
Microsoft
Mozilla
Netscape
ODBC
Policies
Realtek
Redemption
RegisteredApplications
Remedy
Senior Sistemas
Siemens
OpenScape
SoftOla
Trace
UCSolutions

Name	Type	Data
(Default)	REG_SZ	(value not set)
PathFile	REG_SZ	C:\TRACES\Unify\odc\
SciPathFile	REG_SZ	C:\TRACES\Unify\odc\SCITrace.txt
SciTrace	REG_DWORD	0x00000001 (1)
TomsecTrace	REG_DWORD	0x00000000 (0)
Tracelevel	REG_DWORD	0x00000009 (9)
WriteToFile	REG_DWORD	0x00000001 (1)
AudioTrace	REG_DWORD	0x00000001 (1)

Computer\HKEY_CURRENT_USER\Software\Siemens\SoftOla\Trace

Figure 16: AudioTrace key

The WAVE files will monitor and trace all within area marked with green. The trace does not include Windows components, such as Windows Audio API or USB.

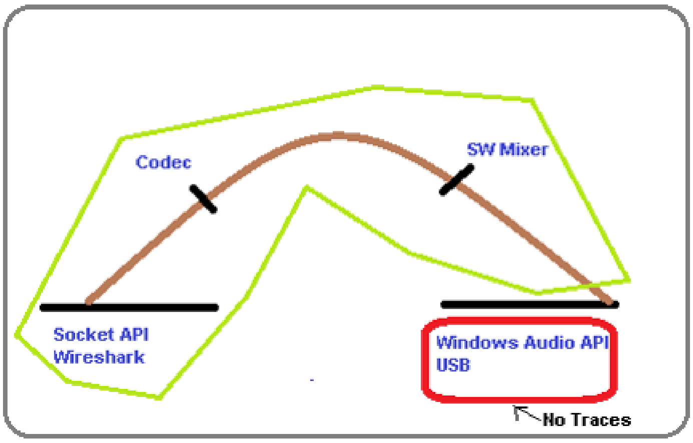


Figure 17: Starting WinDbg

> **NOTICE:** The audio traces only provide information about audio issues. Client traces are still needed for analysis.

3.9.12 PSR Tool

PSR (Problem Steps Recorder) is a free Windows tool that allows for analyzing complex or sporadic scenarios.

The PSR tool is part of the Windows operating system 7, 8.1 and 10.

Follow the instructions below to record and save steps on your computer:

1) To open **Steps Recorder**, click the Windows **Start** button, then select:

Windows Accessories > **Steps Recorder** (in Windows 10),

or

Accessories > **Problem Steps Recorder** (in Windows 7 or Windows 8.1).

2) Click **Start Record**.

3) Reproduce the problem that you are trying to diagnose.

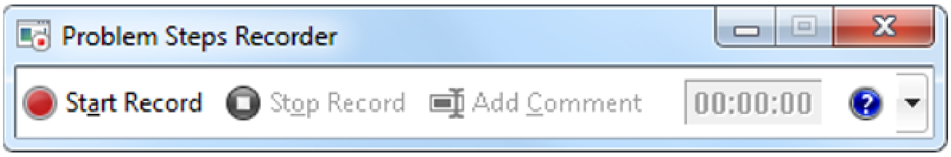
You can pause and resume the recording at any time.

4) Optionally, during recording, you can click **Add Comment**, then use your mouse to select the part of the screen that you want to comment on. Type your comment and then click **OK**.

5) Finally, click **Stop Record**.

6) Review the recording, then click **Save**, choose a name for the ZIP file and the location where you want to save it. Click **Save**.

Attach the ZIP file and send it to the support team that troubleshoots the problem on your PC. The recording can be displayed in any browser .



Follow the instructions below to adjust the settings of the PSR tool:

1) Open **Steps Recorder** and click the down arrow next to the **Help** button. Select **Settings**.

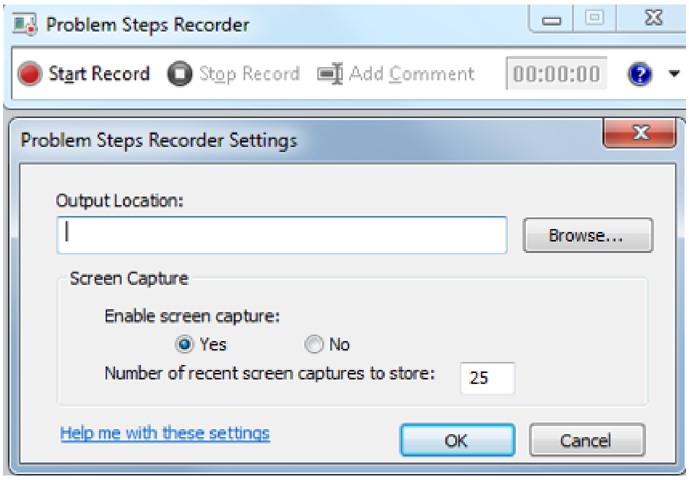
2) You can change the following settings:

- **Output location:** adjust this setting if you do not want to be prompted for a location and file name every time you save a file. F

  or this, select **Browse** to set a default location and file name.
- **Enable screen capture:** select **No** if you do not want to capture screen shots (for example, if your screen displays personal information that you do not want to share.

  The app will still record a text description of your steps.
- **Number of recent screen captures to store:** the default number is 25 screen shots.

  Increase this number if you want to record more screen shots.



Output Location:

Screen Capture

Enable screen capture:

● Yes ○ No

Number of recent screen captures to store: 25

3.10 Memory DUMPs

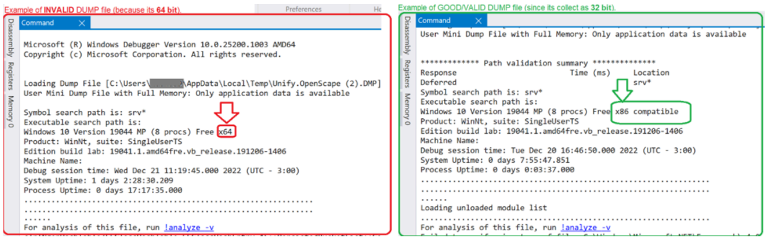
3.10.1 Situations that require memory dumps

In case of crashes (e.g. "Unify.OpenScape.exe stopped working"), dump logs should be collected with the error window or message still displayed. You can close the error window or message once the dump logs are collected.

If the UI is hanging (e.g. UI stops or becomes unresponsive), the dump logs should be collected while the problem is happening, since the application might recover without any user action.

IMPORTANT: The **DUMP** file **MUST** be collect as **32 bit**, not as **64 bit**.

Examples of invalid and valid dump files:



```
Microsoft (R) Windows Debugger Version 10.0.25200.1003 AMD64
Copyright (c) Microsoft Corporation. All rights reserved.


Loading Dump File [C:\Users\[redacted]\AppData\Local\Temp\Unify.OpenScape (2).DMP]
User Mini Dump File with Full Memory: Only application data is available

Symbol search path is: srv*
Executable search path is:
Windows 10 Version 19044 MP (8 procs) Free x64
Product: WinNt, suite: SingleUserTS
Edition build lab: 19041.1.amd64fre.vb_release.191206-1406
Machine Name:
Debug session time: Wed Dec 21 11:19:45.000 2022 (UTC - 3:00)
System Uptime: 1 days 2:28:30.209
Process Uptime: 0 days 17:17:35.000
...............................................................
...............................................................
......
For analysis of this file, run !analyze -v
```

Example of GOOD/VALID DUMP file (since its collect as **32 bit**).
Command
Disassembly
Registers
Memory 0

```
User Mini Dump File with Full Memory: Only application data is available


************* Path validation summary **************
Response                         Time (ms)     Location
Deferred                                       srv*
Symbol search path is: srv*
Executable search path is:
Windows 10 Version 19044 MP (8 procs) Free x86 compatible
Product: WinNt, suite: SingleUserTS
Edition build lab: 19041.1.amd64fre.vb_release.191206-1406
Machine Name:
Debug session time: Tue Dec 20 16:46:50.000 2022 (UTC - 3:00)
System Uptime: 0 days 7:55:47.851
Process Uptime: 0 days 0:03:37.000
................................................................
................................................................
......
Loading unloaded module list
................................................................
For analysis of this file, run !analyze -v
```

These details can be viewed once you open the dump file.

If the **Unify.OpenScape.exe** process memory consumption increases a lot, dump files should be collected more than once. The following steps should be performed in case of high memory usage:

1) Collect a DUMP file just after the application is up and running.

2) Collect additional DUMP files of 200MB-300MB each.

3) Send the files to GVS or Dev teams for further analyze.

Therefore, even if you are using Windows on 64 bits, you must use the 32-bit Task Manager, because the OpenScape (ODC-PE) processes run on 32 bit (except for the TSP module).

NOTICE:

If you can not successfully collect the **DUMP** in **32-bit** format, collect it via **WinDBG** as explained within **chapter 5.3** or **Annex A**.

The following chapters contain information on how to collect dump logs using Windows Task Manager and WinDBG.

3.10.2 Collecting DUMPs via Windows Task Manager (32 bit)

The fastest way to collect DUMPs is via Windows Task Manager (32 bit).

⚠ **WARNING:** If you are using Windows on 64 bits, you must use the 32-bit Task Manager located in the following path:

```
C:\Windows\SysWOW64\taskmgr.exe
```

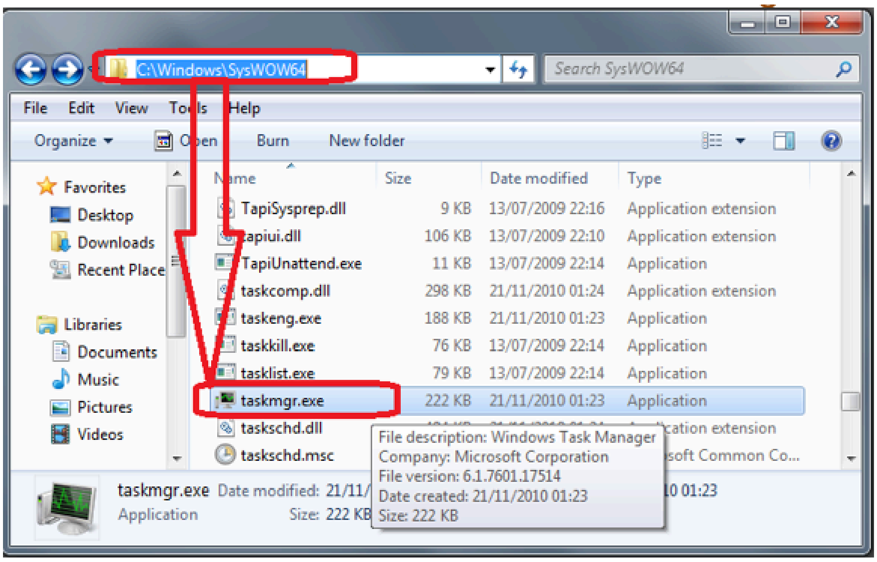
If you are using Windows on 32 bits, then the default Task Manager will be used.

1) Open **Windows Task Manager (32 bit)**.

NOTICE:

In case your operating system is Windows on 64 bits, you need to open the Task Manager on 32 bits. To do so, press simultaneously the **CTRL+SHIFT+ESC** keys.

- Open an Explorer Window.
- Navigate to the following path: `C:\Windows\SysWOW64`.
- Double click on the **TaskMgr.exe** file.



N me	Size	Date modified	Type
TapiSysprep.dll	9 KB	13/07/2009 22:16	Application extension
apiui.dll	106 KB	13/07/2009 22:10	Application extension
TapiUnattend.exe	11 KB	13/07/2009 22:14	Application
taskcomp.dll	298 KB	21/11/2010 01:24	Application extension
taskeng.exe	188 KB	21/11/2010 01:23	Application
taskkill.exe	76 KB	13/07/2009 22:14	Application
tasklist.exe	79 KB	13/07/2009 22:14	Application
taskmgr.exe	222 KB	21/11/2010 01:23	Application
taskschd.dll			cation extension
taskschd.msc			soft Common Co...

File description: Windows Task Manager
Company: Microsoft Corporation
File version: 6.1.7601.17514
Date created: 21/11/2010 01:23
Size: 222 KB

taskmgr.exe Date modified: 21/11/ 10 01:23
Application Size: 222 KB

Figure 18: Running TaskMgr.exe

2) Select the desired process to collect the DUMPs:

In Task Manager (32 bit), locate the desired process for which you want to collect a Dump file.

For example, in case the ODC-PE Client freezes, navigate to the **Processes** tab and right click on the corresponding process and select **Create Dump File**.

“Unify.OpenScape.exe *32”: |“Unify.OpenScape.exe” (“**Details**” tab in **newests** Win10/11):

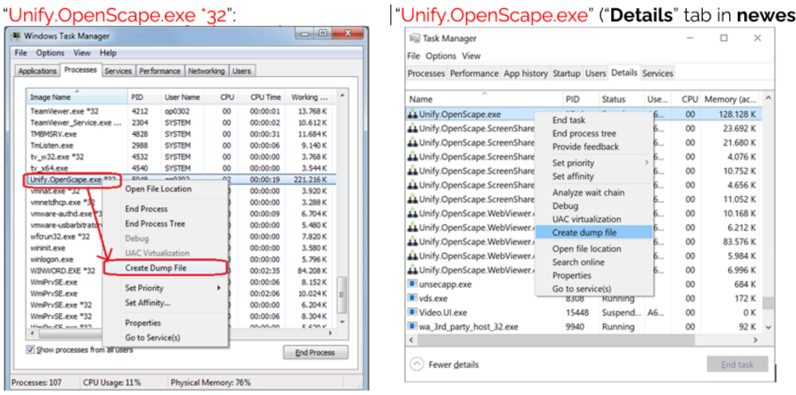
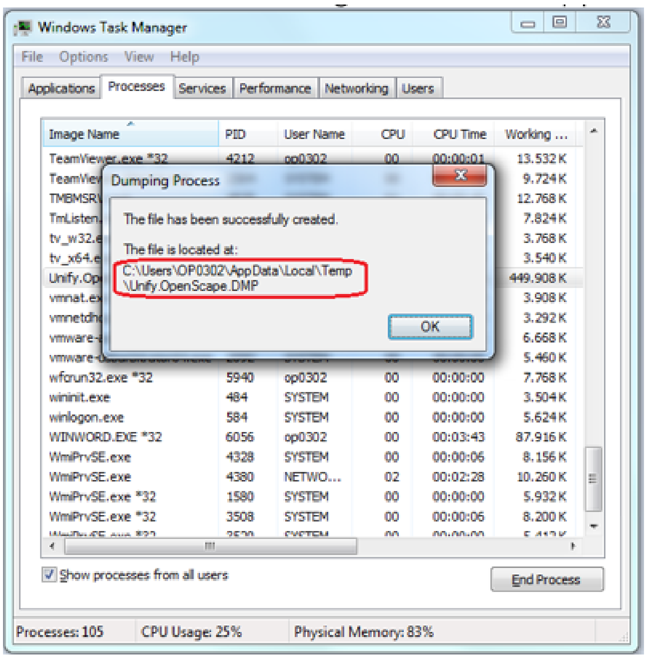


Figure 19: Collecting a Dump file

Wait for the Dump file to be collected.

When ready, a message is displayed containing the Dump file location.

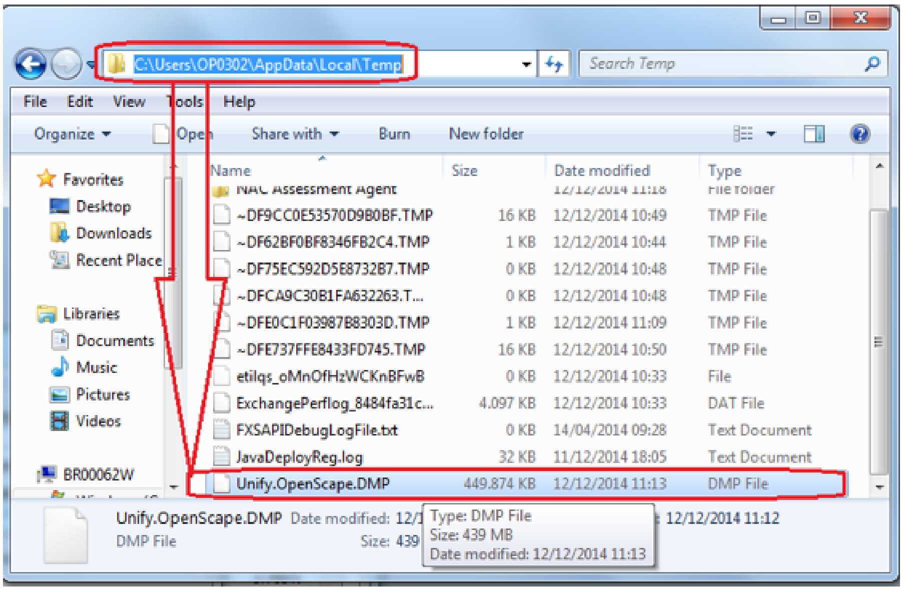

Windows Task Manager
File Options View Help
Applications Processes Services Performance Networking Users
Dumping Process
OK
Show processes from all users
End Process
Processes: 105 CPU Usage: 25% Physical Memory: 83%

The file has been successfully created.

The file is located at:
C:\Users\OP0302\AppData\Local\Temp\Unify.OpenScape.DMP

Figure 20: Dump file location

Navigate to the folder with the Dump file, copy and attach it to the corresponding PRB ticket, along with the PE logs.


C:\Users\OP0302\AppData\Local\Temp
Search Temp
File Edit View Tools Help
Organize Open Share with Burn New folder
Favorites
Desktop
Downloads
Recent Places
Libraries
Documents
Music
Pictures
Videos
BR00062W

Name	Size	Date modified	Type
NAC Assessment Agent		12/12/2014 11:18	File folder
~DF9CC0E53570D9B0BF.TMP	16 KB	12/12/2014 10:49	TMP File
~DF62BF0BF8346FB2C4.TMP	1 KB	12/12/2014 10:44	TMP File
~DF75EC592D5E8732B7.TMP	0 KB	12/12/2014 10:48	TMP File
~DFCA9C30B1FA632263.T...	0 KB	12/12/2014 10:48	TMP File
~DFE0C1F03987B8303D.TMP	1 KB	12/12/2014 11:09	TMP File
~DFE737FFE8433FD745.TMP	16 KB	12/12/2014 10:50	TMP File
etilqs_oMnOfHzWCKnBFwB	0 KB	12/12/2014 10:33	File
ExchangePerflog_8484fa31c...	4.097 KB	12/12/2014 10:33	DAT File
FXSAPIDebugLogFile.txt	0 KB	14/04/2014 09:28	Text Document
JavaDeployReg.log	32 KB	11/12/2014 18:05	Text Document
Unify.OpenScape.DMP	449.874 KB	12/12/2014 11:13	DMP File

Unify.OpenScape.DMP Date modified: 12/1
DMP File Size: 439

Type: DMP File
Size: 439 MB
Date modified: 12/12/2014 11:13

12/12/2014 11:12

Figure 21: Collected Dump file

E.g. `C:\Users\op0302\AppData\Local\Temp\Unify.OpenScape.DMP`

3.10.3 Collecting DUMPs via WinDbg

This chapter describes an overview of collecting Dump files using WinDbg.

NOTICE: If you are using WinDbg for the first time, navigate to Annex A for more detailed instructions.

You can use WinDbg to collect **DUMP 32bit** files. This tool is especially useful in scenarios where ODC-PE chases during the start-up or when it crashes and kills its own process. Using WinDbg, you can activate and collect DUMPs via the command line.

Collecting logs with WinDbg is more complicated than collecting then via Task Manager. It is not installed by default on the Windows operating systems, therefore, it must be first installed. After the installation, you must identify the process ID or name and use commands to complete the DUMP collection.

If you are already familiar with WinDbg, follow the instructions below to collect Dump files:

1) If not installed yet, **install** the "Debugging Tools for Windows (x86)".

   You must use the x86 version even if your operating system is on 64 bits.
2) Start the **PE** Client and wait for it to completely load.
3) Start the WinDbg tool as an administrator.
4) Attach the **process** you are interested in by clicking on **File -> Attach to a Process**, then selecting the "**Unify.OpenScape.exe**" process.
5) This will stop the ODC-PE Client. To resume it, click on **Debug -> Go**.
6) Perform the scenario you want to reproduce: initiate an incoming call, answer it and wait for at least 5 seconds, but not more than 10 seconds.

7) Navigate back to WinDbg and click on **Debug -> Break**. This will stop the ODC-PE Client again.
8) On the **command bar** (displayed in the bottom area), type "`.dump /ma <dump file>`" (e.g. **dump file**: "C:\odc.dump ").
9) **Wait** for the write operation to complete.
10) **Resume** the ODC-PE Client (as described at step 5) and close WinDbg.
11) Provide the **dump file** (approximate size: 400MB to ~1GB) for further analyze.

In case of errors **during ODC-PE start** process (or too fast), you can start the **PE**Client via WinDbg:

1) If not installed yet, **install** the "Debugging Tools for Windows (x86)".

   You must use the x86 version even if your operating system is on 64 bits.
2) Start WinDbg as an administrator.
3) Start the Fusion Client via WinDbg, by clicking on "**File -> Open Executable** " and entering the "**Unify.OpenScape.exe** " process.
4) Perform the scenario you want to reproduce: initiate an incoming call, answer it and wait for at least 5 seconds, but not more than 10 seconds.
5) Navigate back to WinDbg and click on **Debug -> Break**. This will stop the ODC-PE Client..
6) On the **command bar** (displayed in the bottom area), please type "`.dump /ma <dump file>`" (e.g. **dump file**: "C:\odc.dump ")
7) **Wait** for the write operation to complete.
8) **Resume** the ODC (as described at step 5) and close WinDbg.
9) Provide the **dump file** (approximate size: 400MB to ~1GB) for further analyze.

For more information about collecting DUMPs via WinDbg, see Annex A.

3.11 Appendix

3.11.1 Annex A (WinDbg)

This section describes how to collect Dump files using **WinDbg**. It contains information about:

- Where to download **WinDbg**,
- How to install **WinDbg**,
- How to collect Dump files using **WinDbg**.

You can use WinDbg to collect **DUMP 32bit** files. This tool is especially useful in scenarios where ODC-PE chases during the start-up or when it crashes and kills its own process. Using WinDbg, you can activate and collect DUMPs via the command line.

Collecting logs with WinDbg is more complicated than collecting then via Task Manager. It is not installed by default on the Windows operating systems, therefore, it must be first installed. After the installation, you must identify the process ID or name and use commands to complete the DUMP collection.

For quick instructions about collecting DUMPs via **WinDbg**, see chapter Collecting DUMPs via WinDbg on page 36. More detailed instructions are available below:

1) Install **Debugging Tools for Windows (x86)**.

> **NOTICE:** Please note that even on **64-bits** systems, you must install **WinDbgx86** since F4O runs on **32 bits.**

a) Download **Debugging Tools for Windows** via the **Microsoft** official page:

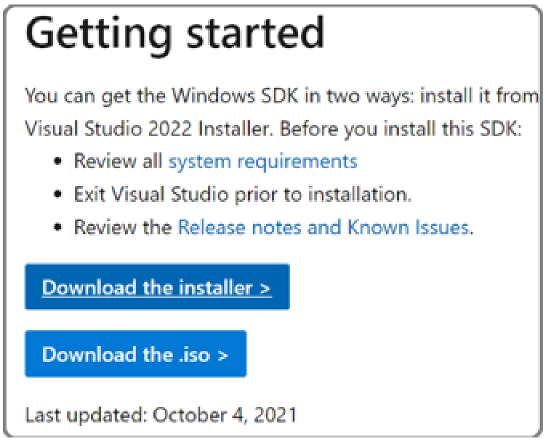
https://developer.microsoft.com/en-us/windows/downloads/windows-sdk/

or

https://msdn.microsoft.com/en-us/windows/hardware/hh852365

> **NOTICE:** If it is not possible to download the tool using the above links, use the following link that contains older SDKs installations:
>
> https://developer.microsoft.com/en-us/windows/downloads/sdk-archive/

b) Click **Download the installer**.



Getting started

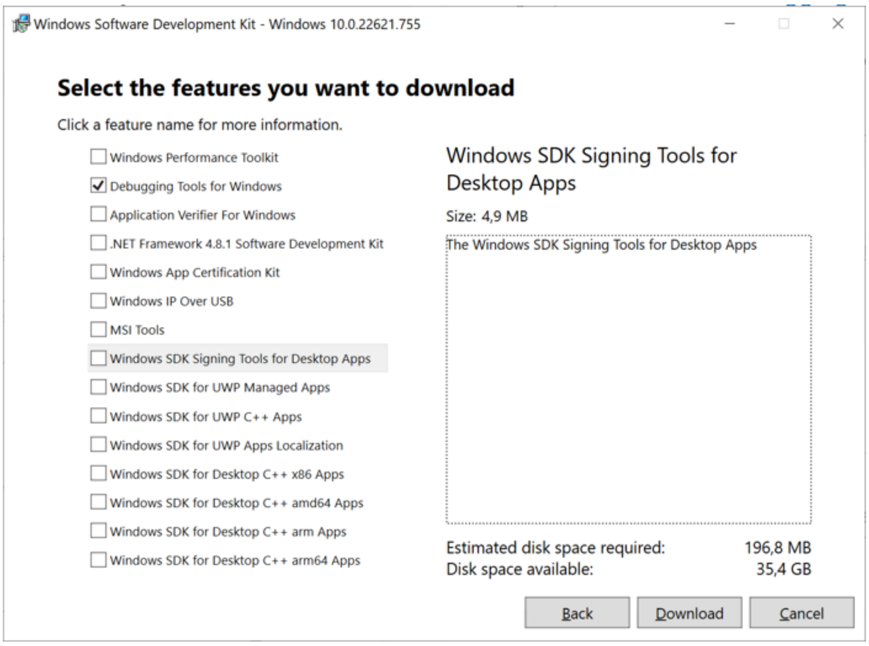
You can get the Windows SDK in two ways: install it from Visual Studio 2022 Installer. Before you install this SDK:

- Review all system requirements
- Exit Visual Studio prior to installation.
- Review the Release notes and Known Issues.

Last updated: October 4, 2021

When the download is ready, run the **winsdksetup.exe** file to install the tool.

c) Select the desired packages to be installed.



Windows Software Development Kit - Windows 10.0.22621.755

Select the features you want to download

Click a feature name for more information.

- □ Windows Performance Toolkit
- ☑ Debugging Tools for Windows
- □ Application Verifier For Windows
- □ .NET Framework 4.8.1 Software Development Kit
- □ Windows App Certification Kit
- □ Windows IP Over USB
- □ MSI Tools
- □ Windows SDK Signing Tools for Desktop Apps
- □ Windows SDK for UWP Managed Apps
- □ Windows SDK for UWP C++ Apps
- □ Windows SDK for UWP Apps Localization
- □ Windows SDK for Desktop C++ x86 Apps
- □ Windows SDK for Desktop C++ amd64 Apps
- □ Windows SDK for Desktop C++ arm Apps
- □ Windows SDK for Desktop C++ arm64 Apps

Windows SDK Signing Tools for Desktop Apps

Size: 4,9 MB

The Windows SDK Signing Tools for Desktop Apps

Estimated disk space required:	196,8 MB
Disk space available:	35,4 GB

d) Click **Download** to start the package installation.

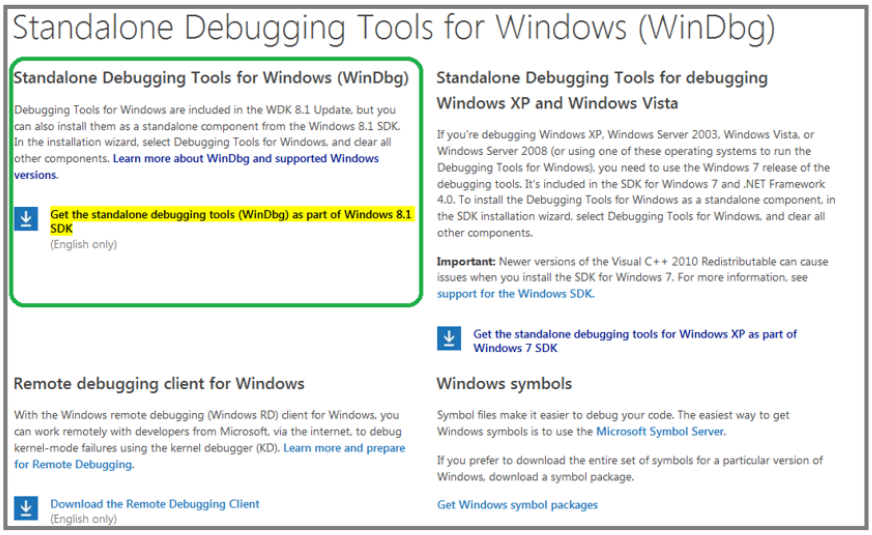
e) Once the package is successfully installed, open Windows search and enter **WeinDbg**.

> **NOTICE:** If you are using **Windows 10 newest SP** or **Windows 11**, continue from step 3.

f) For older SDK installation, you must follow the substeps 1.6 to 2.2. These steps are intended for users with Windows 7, 8.1 or 10 (with old SP).

Select the **Get the standalone debugging tools (WinDbg) as part of Windows 8.1 SDK** or open the following URL with a browser of your choice:

https://www.microsoft.com/click/services/Redirect2.ashx?CR_EAC=300135395



Standalone Debugging Tools for Windows (WinDbg)

Standalone Debugging Tools for Windows (WinDbg)

Debugging Tools for Windows are included in the WDK 8.1 Update, but you can also install them as a standalone component from the Windows 8.1 SDK. In the installation wizard, select Debugging Tools for Windows, and clear all other components. Learn more about WinDbg and supported Windows versions.

Get the standalone debugging tools (WinDbg) as part of Windows 8.1 SDK
(English only)

Standalone Debugging Tools for debugging Windows XP and Windows Vista

If you're debugging Windows XP, Windows Server 2003, Windows Vista, or Windows Server 2008 (or using one of these operating systems to run the Debugging Tools for Windows), you need to use the Windows 7 release of the debugging tools. It's included in the SDK for Windows 7 and .NET Framework 4.0. To install the Debugging Tools for Windows as a standalone component, in the SDK installation wizard, select Debugging Tools for Windows, and clear all other components.

Important: Newer versions of the Visual C++ 2010 Redistributable can cause issues when you install the SDK for Windows 7. For more information, see support for the Windows SDK.

Get the standalone debugging tools for Windows XP as part of Windows 7 SDK

Remote debugging client for Windows

With the Windows remote debugging (Windows RD) client for Windows, you can work remotely with developers from Microsoft, via the internet, to debug kernel-mode failures using the kernel debugger (KD). Learn more and prepare for Remote Debugging.

Download the Remote Debugging Client
(English only)

Windows symbols

Symbol files make it easier to debug your code. The easiest way to get Windows symbols is to use the Microsoft Symbol Server.

If you prefer to download the entire set of symbols for a particular version of Windows, download a symbol package.

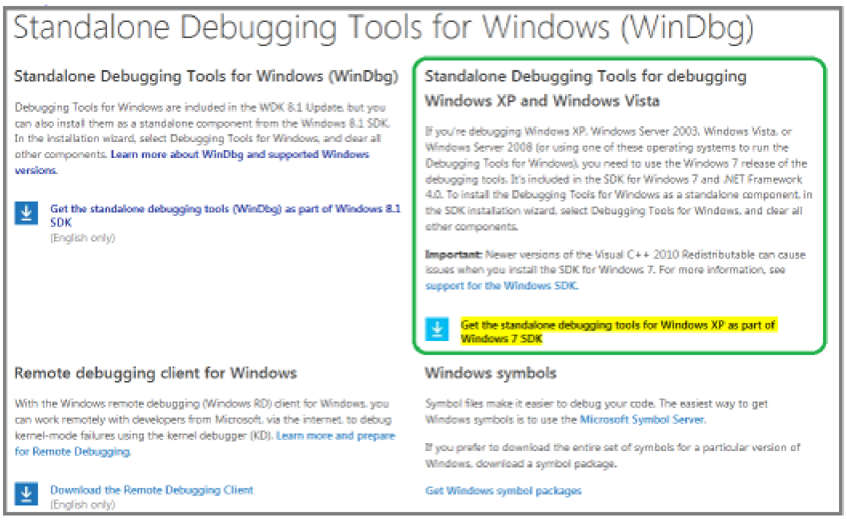
Get Windows symbol packages

Once the download is ready, run the **sdksetup.exe** file to start the installation of WinDbg.

g) If the **Debugging Tools for Windows (x86)** for **Windows 10** package does **not** work (e.g. on several variations of **Windows 7**), you might need to install the specific package for it.

Therefore, click on **Get the standalone debugging tools for Windows XP as part of Windows 7 SDK** or open the following URL with a browser of your choice:

https://www.microsoft.com/en-us/download/confirmation.aspx?id=8279



Standalone Debugging Tools for Windows (WinDbg)

Standalone Debugging Tools for Windows (WinDbg)

Debugging Tools for Windows are included in the WDK 8.1 Update, but you can also install them as a standalone component from the Windows 8.1 SDK. In the installation wizard, select Debugging Tools for Windows, and clear all other components. Learn more about WinDbg and supported Windows versions.

Get the standalone debugging tools (WinDbg) as part of Windows 8.1 SDK
(English only)

Standalone Debugging Tools for debugging Windows XP and Windows Vista

If you're debugging Windows XP, Windows Server 2003, Windows Vista, or Windows Server 2008 (or using one of these operating systems to run the Debugging Tools for Windows), you need to use the Windows 7 release of the debugging tools. It's included in the SDK for Windows 7 and .NET Framework 4.0. To install the Debugging Tools for Windows as a standalone component, in the SDK installation wizard, select Debugging Tools for Windows, and clear all other components.

Important: Newer versions of the Visual C++ 2010 Redistributable can cause issues when you install the SDK for Windows 7. For more information, see support for the Windows SDK.

Get the standalone debugging tools for Windows XP as part of Windows 7 SDK

Remote debugging client for Windows

With the Windows remote debugging (Windows RD) client for Windows, you can work remotely with developers from Microsoft, via the internet, to debug kernel-mode failures using the kernel debugger (KD). Learn more and prepare for Remote Debugging.

Download the Remote Debugging Client
(English only)

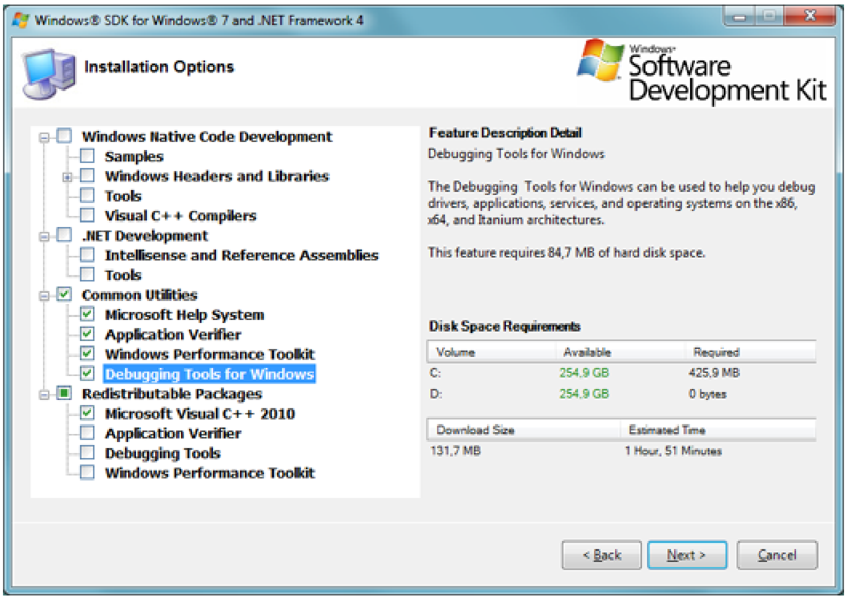
Windows symbols

Symbol files make it easier to debug your code. The easiest way to get Windows symbols is to use the Microsoft Symbol Server.

If you prefer to download the entire set of symbols for a particular version of Windows, download a symbol package.

Get Windows symbol packages

Once the download is ready, run the **winsdk_web.exe** file to start the installation of WinDbg.



Windows® SDK for Windows® 7 and .NET Framework 4

Installation Options

Windows Software Development Kit

- □ Windows Native Code Development
  - □ Samples
  - □ Windows Headers and Libraries
  - □ Tools
  - □ Visual C++ Compilers
- □ .NET Development
  - □ Intellisense and Reference Assemblies
  - □ Tools
- ☑ Common Utilities
  - ☑ Microsoft Help System
  - ☑ Application Verifier
  - ☑ Windows Performance Toolkit
  - ☑ Debugging Tools for Windows
- ■ Redistributable Packages
  - ☑ Microsoft Visual C++ 2010
  - □ Application Verifier
  - □ Debugging Tools
  - □ Windows Performance Toolkit

Feature Description Detail
Debugging Tools for Windows

The Debugging Tools for Windows can be used to help you debug drivers, applications, services, and operating systems on the x86, x64, and Itanium architectures.

This feature requires 84,7 MB of hard disk space.

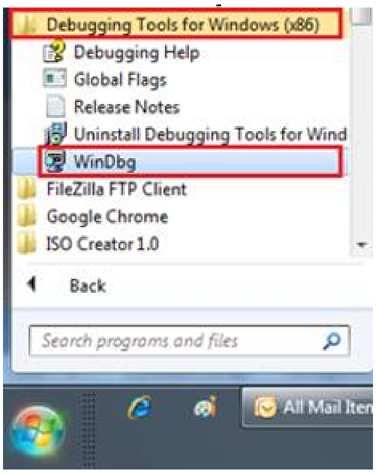
Disk Space Requirements

Volume	Available	Required
C:	254,9 GB	425,9 MB
D:	254,9 GB	0 bytes

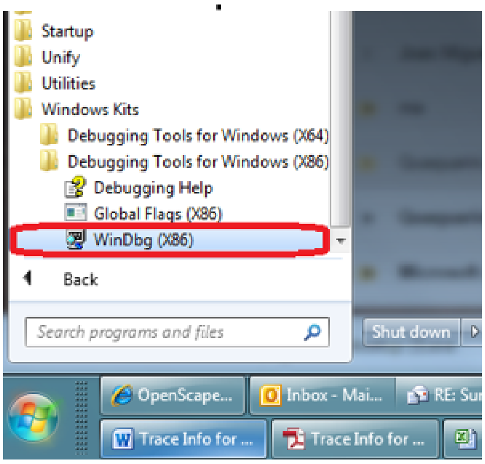
Download Size	Estimated Time
131,7 MB	1 Hour, 51 Minutes

2) Start the **WinDbg** tool.

Example 1:

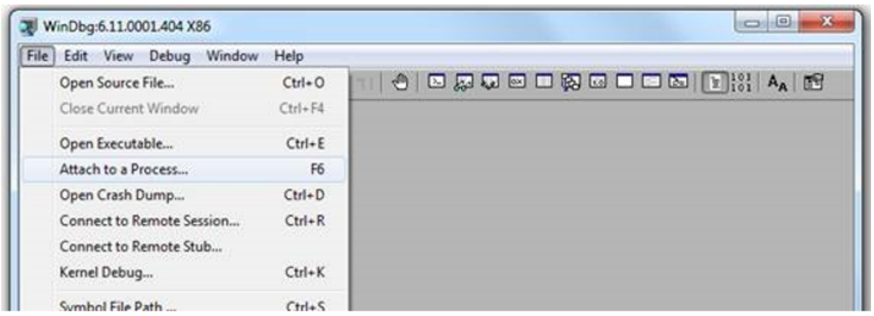


Example 2:

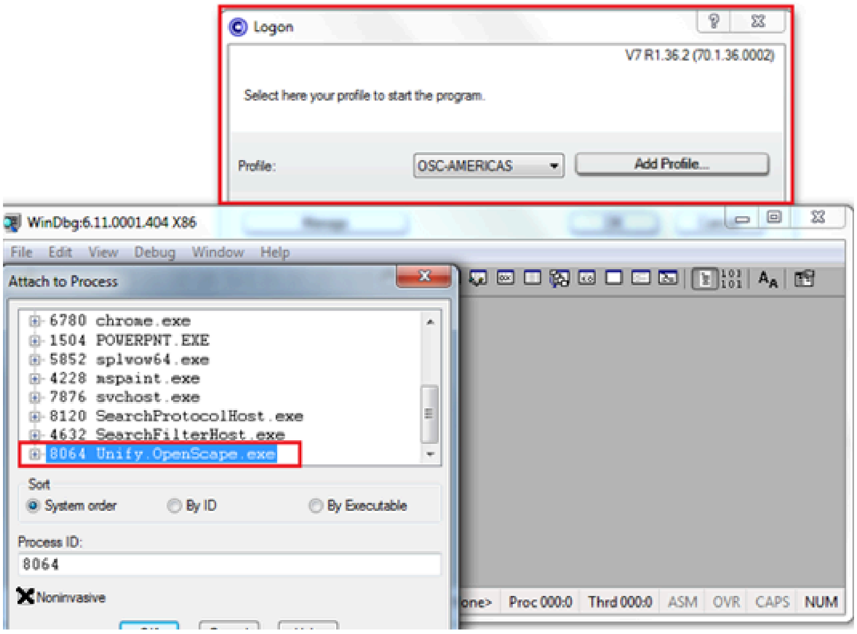


3) Attach the ODC process.

a) If you need a DUMP file from an already running ODC, then on WinDbg click **File -> Attach to a Process** and locate the **Unify.OpenScape.exe** process.


WinDbg:6.11.0001.404 X86
File Edit View Debug Window Help
Open Source File... Ctrl+O
Close Current Window Ctrl+F4
Open Executable... Ctrl+E
Attach to a Process... F6
Open Crash Dump... Ctrl+D
Connect to Remote Session... Ctrl+R
Connect to Remote Stub...
Kernel Debug... Ctrl+K
Symbol File Path ... Ctrl+S

Next, select the desired process to collect the DUMP file.


Logon
V7 R1.36.2 (70.1.36.0002)
Add Profile...
WinDbg:6.11.0001.404 X86
File Edit View Debug Window Help
Attach to Process
one> Proc 000:0 Thrd 000:0 ASM OVR CAPS NUM

Select here your profile to start the program.

Profile: OSC-AMERICAS

```
6780 chrome.exe
1504 POWERPNT.EXE
5852 splwow64.exe
4228 mspaint.exe
7876 svchost.exe
8120 SearchProtocolHost.exe
4632 SearchFilterHost.exe
8064 Unify.OpenScape.exe
```

Sort: ● System order ○ By ID ○ By Executable

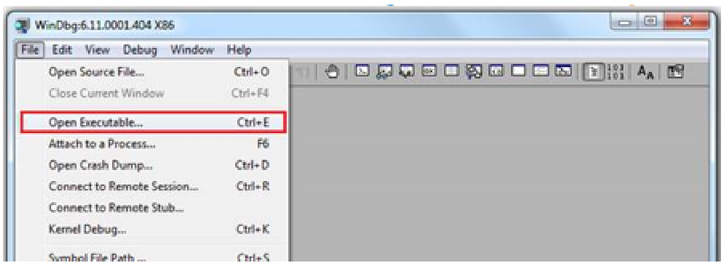
Process ID: 8064

☒ Noninvasive

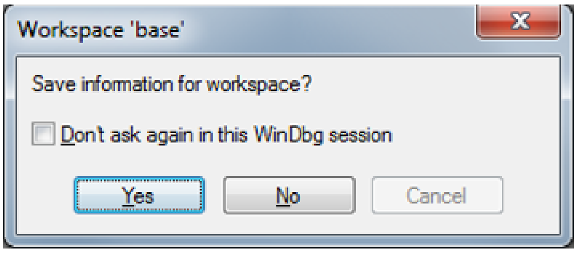
Check the **Noninvasive** option and click **OK**.

Continue with step c).

b) If you need a DUMP file from before ODC starts, then on WinDbg click **File -> Open Executable** and locate the **Unify.OpenScape.exe** process.


WinDbg:6.11.0001.404 X86
File Edit View Debug Window Help
Open Source File... Ctrl+O
Close Current Window Ctrl+F4
Open Executable... Ctrl+E
Attach to a Process... F6
Open Crash Dump... Ctrl+D
Connect to Remote Session... Ctrl+R
Connect to Remote Stub...
Kernel Debug... Ctrl+K
Symbol File Path ... Ctrl+S

Click **Yes** in the pop-up window displayed.

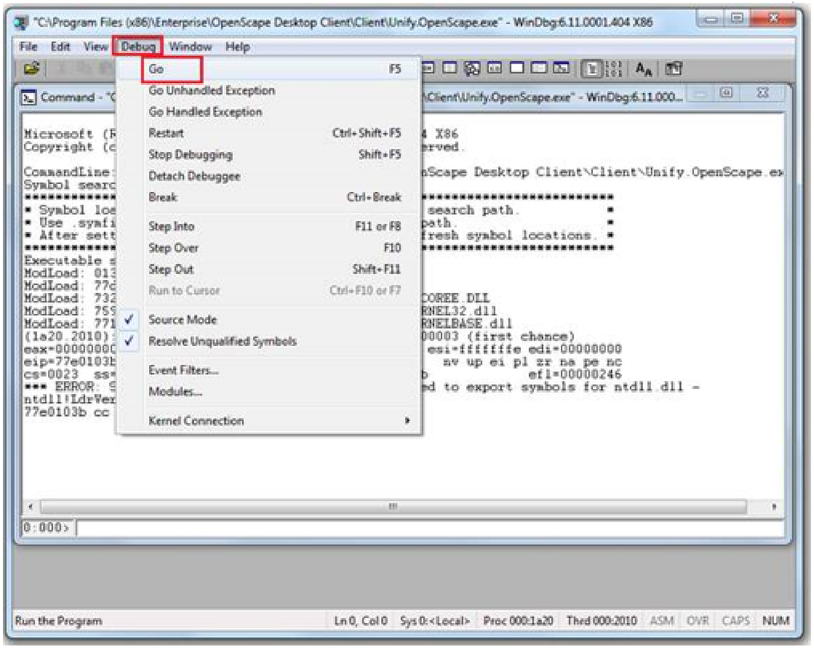


Workspace 'base'

Save information for workspace?

□ Don't ask again in this WinDbg session

This stops the ODC Client. To resume it, click **Debug -> Go**.



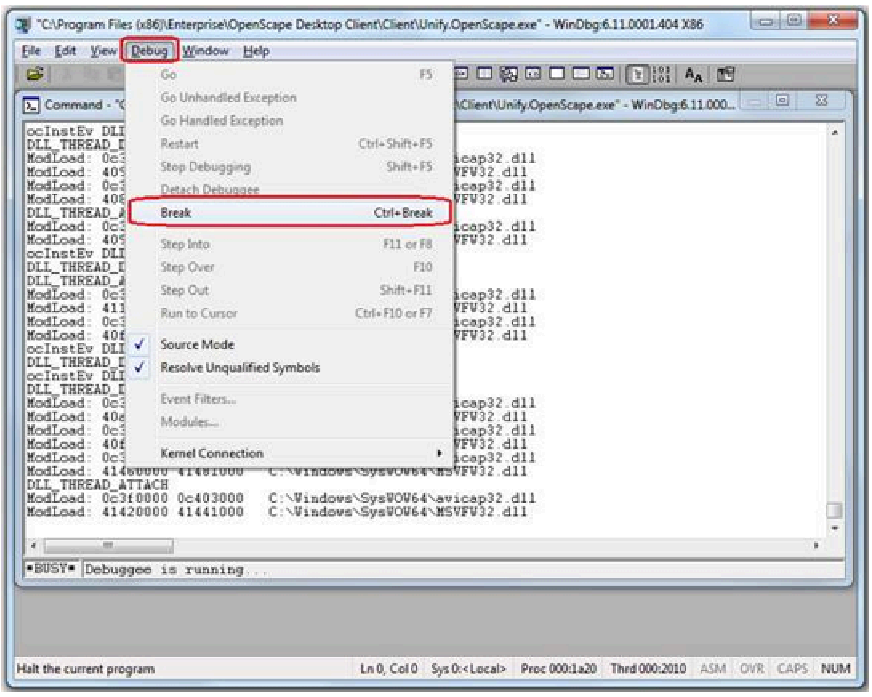
```
Microsoft (R                                              4 X86
Copyright (c                                              erved.

CommandLine:                                              nScape Desktop Client\Client\Unify.OpenScape.ex
Symbol searc
************                                              **************************
* Symbol loc                                               search path.            *
* Use .symfi                                              path.                    *
* After sett                                              fresh symbol locations.  *
************                                              **************************
Executable s
ModLoad: 013
ModLoad: 77c
ModLoad: 732                                              COREE.DLL
ModLoad: 75S                                              RNEL32.dll
ModLoad: 771                                              RNELBASE.dll
(1a20.2010):                                              00003 (first chance)
eax=0000000C                                               esi=fffffffe edi=00000000
eip=77e0103b                                                 nv up ei pl zr na pe nc
cs=0023  ss=                                              b            efl=00000246
*** ERROR: S                                              ed to export symbols for ntdll.dll -
ntdll!LdrVer
77e0103b cc
```

c) Perform the scenario to reproduce the error and collect the DUMP file.

d) This steps must be followed only if step b) applies for you.

   If step a) applies for you, you must skip to step e).

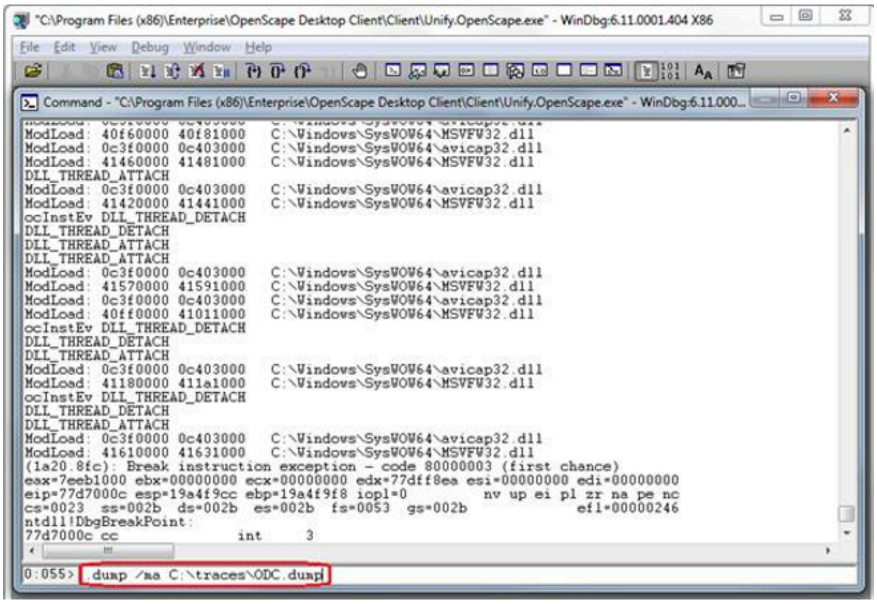
   Once the error occurs, open WinDbg again and click on **Debug -> Break**. This action will stop the ODC Client again.



```
DLL_THREAD_ATTACH
ModLoad: 0c3f0000 0c403000   C:\Windows\SysWOW64\avicap32.dll
ModLoad: 41420000 41441000   C:\Windows\SysWOW64\MSVFW32.dll
```

```
*BUSY* Debuggee is running...
```

e) Type `.dump /ma <dump file>` in the command bar, displayed at the bottom of the WinDbg window. For example, you can use `C:\odc.dump`as "dump file".

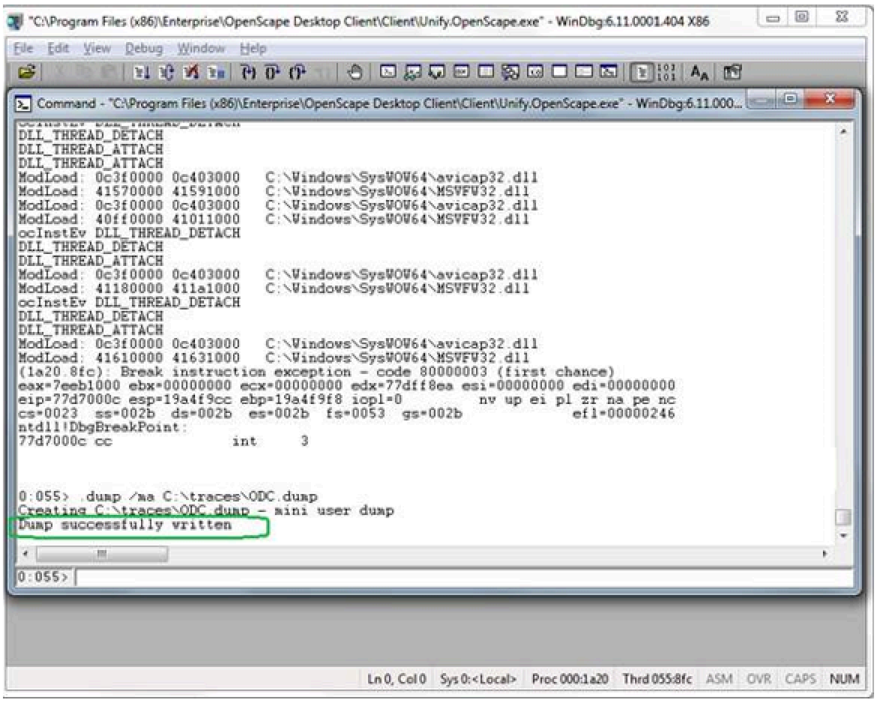

"C:\Program Files (x86)\Enterprise\OpenScape Desktop Client\Client\Unify.OpenScape.exe" - WinDbg:6.11.0001.404 X86
File Edit View Debug Window Help
Command - "C:\Program Files (x86)\Enterprise\OpenScape Desktop Client\Client\Unify.OpenScape.exe" - WinDbg:6.11.000...

```
ModLoad: 40f60000 40f81000   C:\Windows\SysWOW64\MSVFW32.dll
ModLoad: 0c3f0000 0c403000   C:\Windows\SysWOW64\avicap32.dll
ModLoad: 41460000 41481000   C:\Windows\SysWOW64\MSVFW32.dll
DLL_THREAD_ATTACH
ModLoad: 0c3f0000 0c403000   C:\Windows\SysWOW64\avicap32.dll
ModLoad: 41420000 41441000   C:\Windows\SysWOW64\MSVFW32.dll
ocInstEv DLL_THREAD_DETACH
DLL_THREAD_DETACH
DLL_THREAD_ATTACH
DLL_THREAD_ATTACH
ModLoad: 0c3f0000 0c403000   C:\Windows\SysWOW64\avicap32.dll
ModLoad: 41570000 41591000   C:\Windows\SysWOW64\MSVFW32.dll
ModLoad: 0c3f0000 0c403000   C:\Windows\SysWOW64\avicap32.dll
ModLoad: 40ff0000 41011000   C:\Windows\SysWOW64\MSVFW32.dll
ocInstEv DLL_THREAD_DETACH
DLL_THREAD_DETACH
DLL_THREAD_ATTACH
ModLoad: 0c3f0000 0c403000   C:\Windows\SysWOW64\avicap32.dll
ModLoad: 41180000 411a1000   C:\Windows\SysWOW64\MSVFW32.dll
ocInstEv DLL_THREAD_DETACH
DLL_THREAD_DETACH
DLL_THREAD_ATTACH
ModLoad: 0c3f0000 0c403000   C:\Windows\SysWOW64\avicap32.dll
ModLoad: 41610000 41631000   C:\Windows\SysWOW64\MSVFW32.dll
(1a20.8fc): Break instruction exception - code 80000003 (first chance)
eax=7eeb1000 ebx=00000000 ecx=00000000 edx=77dff8ea esi=00000000 edi=00000000
eip=77d7000c esp=19a4f9cc ebp=19a4f9f8 iopl=0         nv up ei pl zr na pe nc
cs=0023  ss=002b  ds=002b  es=002b  fs=0053  gs=002b             efl=00000246
ntdll!DbgBreakPoint:
77d7000c cc              int     3
```

```
0:055> .dump /ma C:\traces\ODC.dump
```

f) **Wait** for the **write operation** to be complete. Depending on the size of the DUMP file, this action might take several minutes.

g) Once the write operation is completed, the message **Dump successfully written** is displayed. Next, close both ODC Client and WinDbg.

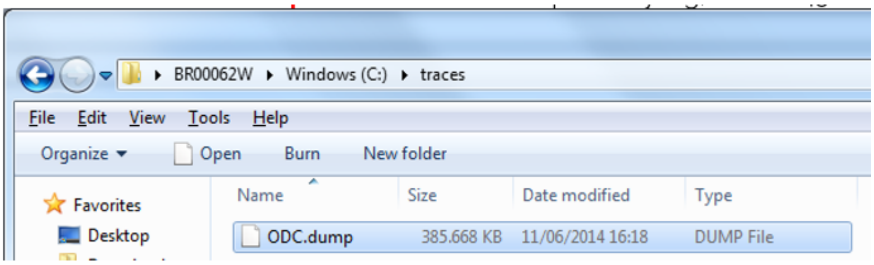


"C:\Program Files (x86)\Enterprise\OpenScape Desktop Client\Client\Unify.OpenScape.exe" - WinDbg:6.11.0001.404 X86
File Edit View Debug Window Help
Command - "C:\Program Files (x86)\Enterprise\OpenScape Desktop Client\Client\Unify.OpenScape.exe" - WinDbg:6.11.000...

```
DLL_THREAD_DETACH
DLL_THREAD_ATTACH
DLL_THREAD_ATTACH
ModLoad: 0c3f0000 0c403000   C:\Windows\SysWOW64\avicap32.dll
ModLoad: 41570000 41591000   C:\Windows\SysWOW64\MSVFW32.dll
ModLoad: 0c3f0000 0c403000   C:\Windows\SysWOW64\avicap32.dll
ModLoad: 40ff0000 41011000   C:\Windows\SysWOW64\MSVFW32.dll
ocInstEv DLL_THREAD_DETACH
DLL_THREAD_DETACH
DLL_THREAD_ATTACH
ModLoad: 0c3f0000 0c403000   C:\Windows\SysWOW64\avicap32.dll
ModLoad: 41180000 411a1000   C:\Windows\SysWOW64\MSVFW32.dll
ocInstEv DLL_THREAD_DETACH
DLL_THREAD_DETACH
DLL_THREAD_ATTACH
ModLoad: 0c3f0000 0c403000   C:\Windows\SysWOW64\avicap32.dll
ModLoad: 41610000 41631000   C:\Windows\SysWOW64\MSVFW32.dll
(1a20.8fc): Break instruction exception - code 80000003 (first chance)
eax=7eeb1000 ebx=00000000 ecx=00000000 edx=77dff8ea esi=00000000 edi=00000000
eip=77d7000c esp=19a4f9cc ebp=19a4f9f8 iopl=0         nv up ei pl zr na pe nc
cs=0023  ss=002b  ds=002b  es=002b  fs=0053  gs=002b             efl=00000246
ntdll!DbgBreakPoint:
77d7000c cc              int     3


0:055> .dump /ma C:\traces\ODC.dump
Creating C:\traces\ODC.dump - mini user dump
Dump successfully written

0:055>
```

h) Provide the Dump file collected to GVS for further analysis.

The size of the Dump file is around 450 MB, as displayed in the example below.



Name	Size	Date modified	Type
ODC.dump	385.668 KB	11/06/2014 16:18	DUMP File

NOTICE: Please note that the folder with the Dump file is the one defined in step e).

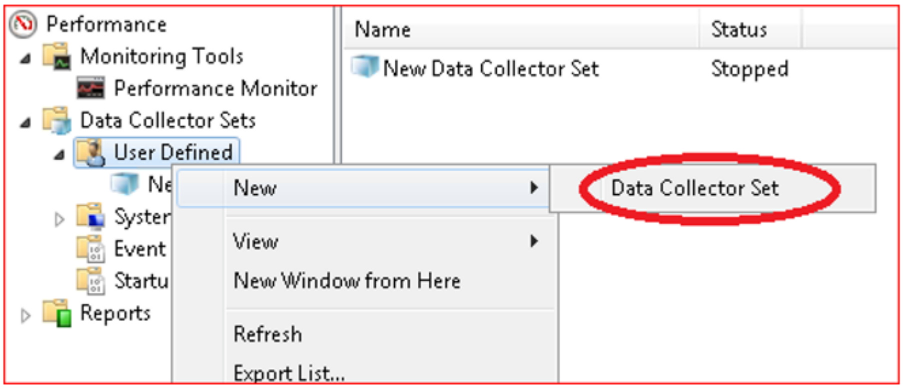
3.11.2 Annex B (perfmon)

This section describes how to collect logs using **MS Performance Monitor**. These logs might be necessary for further analyzing the operating system (Windows) behaviour and the status of specific applications.

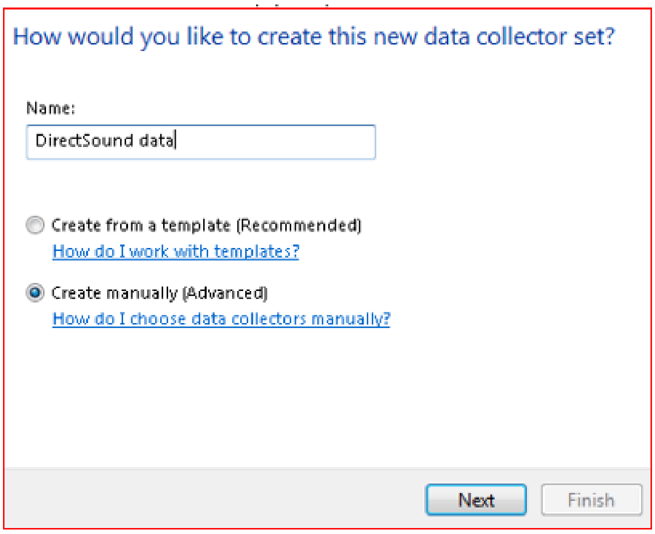
Follow the steps below to collect logs with **MS Performance Monitor**:

1) Click the Windows Start button and search for **perfmon**. Select **perfmon.exe**.

2) Create a new **Data Collector Set** by navigating to **Data Collector Sets \ User Defined** in the left hand panel. Right click on **User Defined**, then select **New \ Data Collector Set** from the context menu.



3) Enter a custom name for the **Data Collector Set** in the **Name** field, then click **Next**.

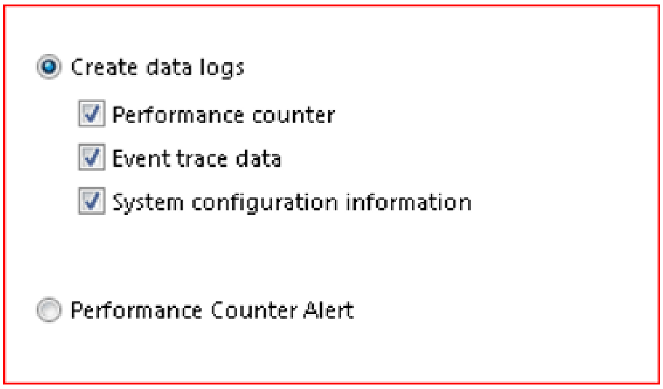

How would you like to create this new data collector set?

Name:
DirectSound data

○ Create from a template (Recommended)
How do I work with templates?

● Create manually (Advanced)
How do I choose data collectors manually?

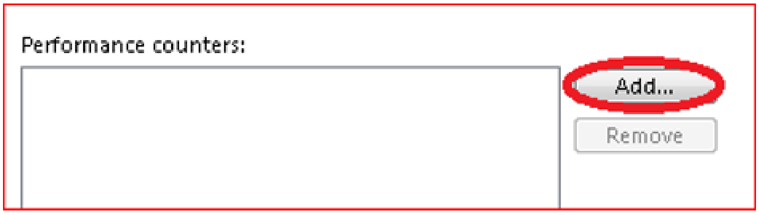
4) Select **Create Data Logs**, then check the following option: **Performance counter, Event trace data, System configuration information**. Click **Next**.


● Create data logs

☑ Performance counter
☑ Event trace data
☑ System configuration information

○ Performance Counter Alert

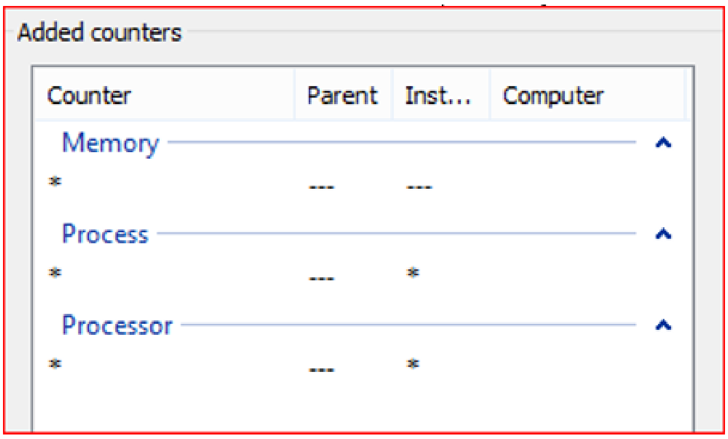
5) Click **Add**.



6) Add the following counters:

- **Process** - <All instances>
- **Processor** - <All instances>
- **Memory**

In the **Added counters** panel, you should now see:

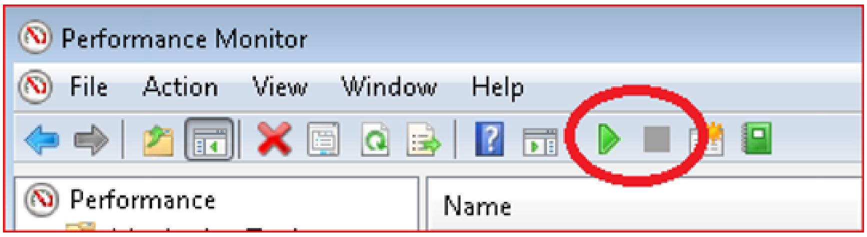

Added counters

Counter	Parent	Inst...	Computer
Memory			
*	---	---	
Process			
*	---	*	
Processor			
*	---	*	

7) Click **OK**, then click **Finish**, to create the data set.

8) Before running the data set, right click on it, select **Properties**. Set the desired output location on the **Directory**, and click **OK**.

9) You can now start or stop the logs by selecting the data set and clicking the play/ stop buttons on the menu bar.



3.11.3 Annex C (TSP logs)

This section describes how to collect **TSP** traces on **Windows 7 and higher**

1) Activate the **regedit key** that corresponds to the PBX used:

   - For **H3K** (or **HP2K**, **OSO**, or any other PBX **except for the HP4K family)**:

     Create the Registry value **DBGTrace**"=dword:**00000001** in the following registry key:
     [HKEY_LOCAL_MACHINE\SOFTWARE\Siemens\optiClientTSP Hicom 150\Trace]

     If requested by GVS or Dev teams, add the following registry:
     `Value "DBGTraceEnh"=dword:00000001 in the same registry key.`

   - For **H4K**:

     Create the Registry – Value "**DBGTrace**"=dword:**00000001** in the following registry key:
     [HKEY_LOCAL_MACHINE\SOFTWARE\Siemens\optiClientTSP HiPath 4000 V2\Trace]

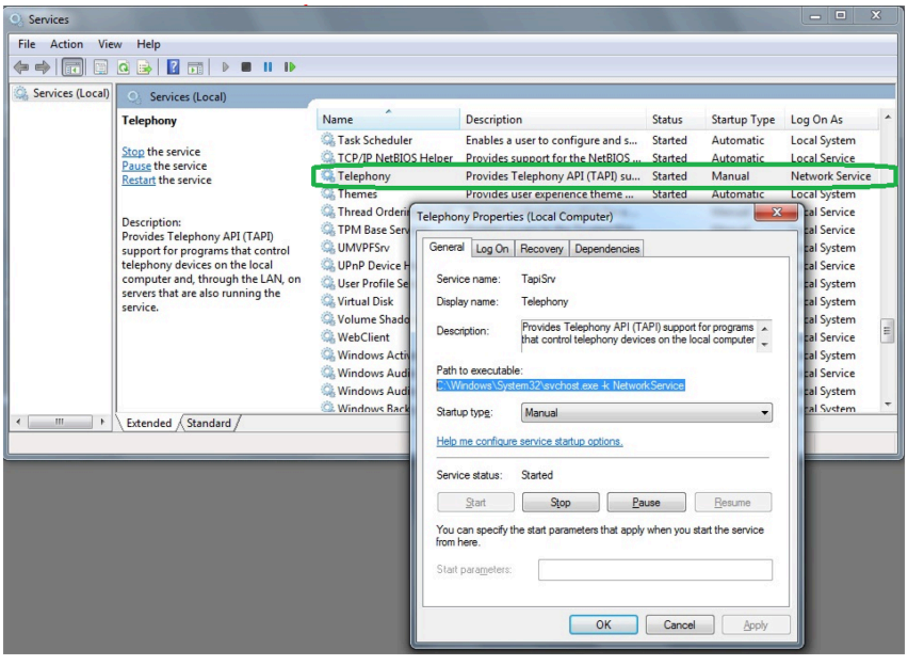
     If requested by GVS or Dev teams, add the following registry:
     `Value "DBGTraceEnh"=dword:00000001in the same registry key.`

     To activate the trace you must restart the **telephony** service or, if it is not possible, **restart the pc**.

2) Check if the **Telephony** service is up and running:

   - Open Windows Services: **Windows Start - Run… - services.msc**
   - Check if the **Telephony** service is running.
   - Ensure the path to the executable is

     ```
     C:\Windows\System32\svchost.exe -k NetworkService
     ```



Telephony

Stop the service
Pause the service
Restart the service

Description:
Provides Telephony API (TAPI) support for programs that control telephony devices on the local computer and, through the LAN, on servers that are also running the service.

Name	Description	Status	Startup Type	Log On As
Task Scheduler	Enables a user to configure and s...	Started	Automatic	Local System
TCP/IP NetBIOS Helper	Provides support for the NetBIOS ...	Started	Automatic	Local Service
Telephony	Provides Telephony API (TAPI) su...	Started	Manual	Network Service
Themes	Provides user experience theme ...	Started	Automatic	Local System
Thread Orderi				al Service
TPM Base Serv				al Service
UMVPFSrv				al System
UPnP Device H				al Service
User Profile Se				al System
Virtual Disk				al System
Volume Shado				al System
WebClient				al Service
Windows Acti				al System
Windows Audi				al Service
Windows Audi				al System
Windows Back				al System

Service name: TapiSrv

Display name: Telephony

Description: Provides Telephony API (TAPI) support for programs that control telephony devices on the local computer

Path to executable:
C:\Windows\System32\svchost.exe -k NetworkService

Startup type: Manual

Help me configure service startup options.

Service status: Started

You can specify the start parameters that apply when you start the service from here.

Start parameters:

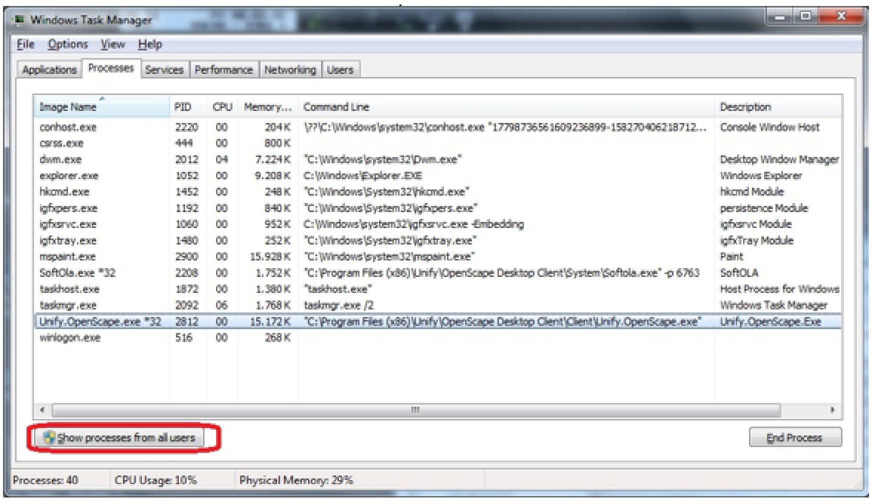
Figure 22: Windows Telephony service

3) Identify the specific **svchost.exe** process with the **Telephony service** on it in one of the following ways:

- Via Windows **Task Manager (recommended)**

  Open Window **Task Manager**.

  Select to show process from **all Users**.



Windows Task Manager
File Options View Help
Applications Processes Services Performance Networking Users

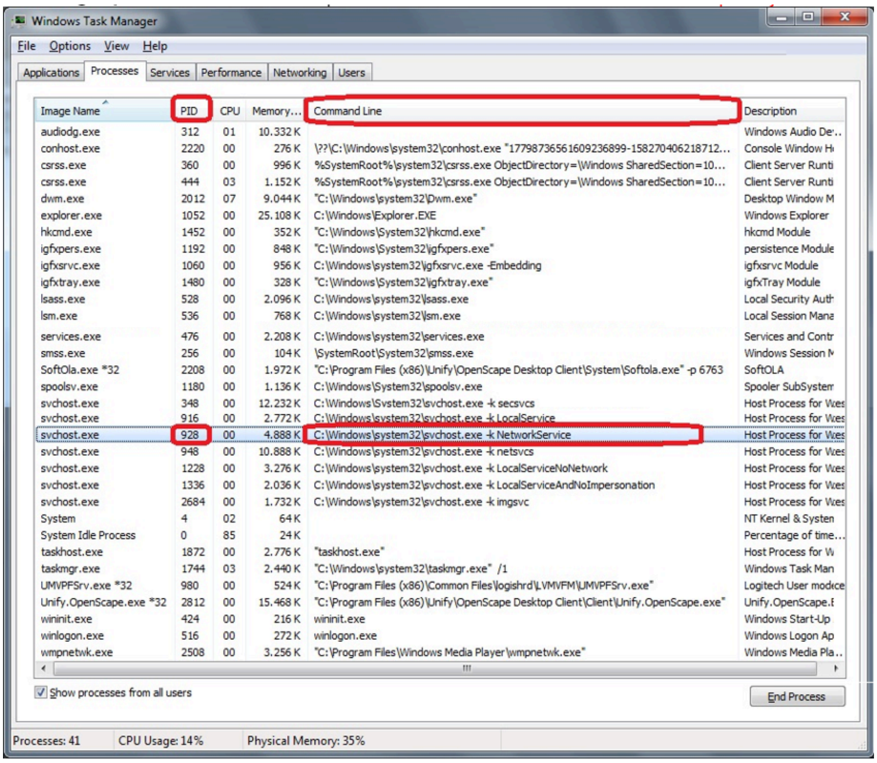
Image Name	PID	CPU	Memory…	Command Line	Description
conhost.exe	2220	00	204 K	\??\C:\Windows\system32\conhost.exe "1779873656 1609236899-158270406218712…	Console Window Host
csrss.exe	444	00	800 K		
dwm.exe	2012	04	7.224 K	"C:\Windows\system32\Dwm.exe"	Desktop Window Manager
explorer.exe	1052	00	9.208 K	C:\Windows\Explorer.EXE	Windows Explorer
hkcmd.exe	1452	00	248 K	"C:\Windows\System32\hkcmd.exe"	hkcmd Module
igfxpers.exe	1192	00	840 K	"C:\Windows\System32\igfxpers.exe"	persistence Module
igfxsrvc.exe	1060	00	952 K	C:\Windows\system32\igfxsrvc.exe -Embedding	igfxsrvc Module
igfxtray.exe	1480	00	252 K	"C:\Windows\System32\igfxtray.exe"	igfxTray Module
mspaint.exe	2900	00	15.928 K	"C:\Windows\system32\mspaint.exe"	Paint
SoftOla.exe *32	2208	00	1.752 K	"C:\Program Files (x86)\Unify\OpenScape Desktop Client\System\Softola.exe" -p 6763	SoftOLA
taskhost.exe	1872	00	1.380 K	"taskhost.exe"	Host Process for Windows
taskmgr.exe	2092	06	1.768 K	taskmgr.exe /2	Windows Task Manager
Unify.OpenScape.exe *32	2812	00	15.172 K	"C:\Program Files (x86)\Unify\OpenScape Desktop Client\Client\Unify.OpenScape.exe"	Unify.OpenScape.Exe
winlogon.exe	516	00	268 K		

Show processes from all users
End Process
Processes: 40 CPU Usage: 10% Physical Memory: 29%

Figure 23: Windows Task Manager

Select to show the **PID** and **Command Line**.

Check the **PID** from process with same Command Line as the Telephony service:



Windows Task Manager
File Options View Help
Applications Processes Services Performance Networking Users

Image Name	PID	CPU	Memory…	Command Line	Description
audiodg.exe	312	01	10.332 K		Windows Audio De'..
conhost.exe	2220	00	276 K	\??\C:\Windows\system32\conhost.exe "1779873656 1609236899-158270406218712…	Console Window H
csrss.exe	360	00	996 K	%SystemRoot%\system32\csrss.exe ObjectDirectory=\Windows SharedSection=10…	Client Server Runti
csrss.exe	444	03	1.152 K	%SystemRoot%\system32\csrss.exe ObjectDirectory=\Windows SharedSection=10…	Client Server Runti
dwm.exe	2012	07	9.044 K	"C:\Windows\system32\Dwm.exe"	Desktop Window M
explorer.exe	1052	00	25.108 K	C:\Windows\Explorer.EXE	Windows Explorer
hkcmd.exe	1452	00	352 K	"C:\Windows\System32\hkcmd.exe"	hkcmd Module
igfxpers.exe	1192	00	848 K	"C:\Windows\System32\igfxpers.exe"	persistence Module
igfxsrvc.exe	1060	00	956 K	C:\Windows\system32\igfxsrvc.exe -Embedding	igfxsrvc Module
igfxtray.exe	1480	00	328 K	"C:\Windows\System32\igfxtray.exe"	igfxTray Module
lsass.exe	528	00	2.096 K	C:\Windows\system32\lsass.exe	Local Security Auth
lsm.exe	536	00	768 K	C:\Windows\system32\lsm.exe	Local Session Mana
services.exe	476	00	2.208 K	C:\Windows\system32\services.exe	Services and Contr
smss.exe	256	00	104 K	\SystemRoot\System32\smss.exe	Windows Session M
SoftOla.exe *32	2208	00	1.972 K	"C:\Program Files (x86)\Unify\OpenScape Desktop Client\System\Softola.exe" -p 6763	SoftOLA
spoolsv.exe	1180	00	1.136 K	C:\Windows\System32\spoolsv.exe	Spooler SubSystem
svchost.exe	348	00	12.232 K	C:\Windows\Svstem32\svchost.exe -k secsvcs	Host Process for Wes
svchost.exe	916	00	2.772 K	C:\Windows\system32\svchost.exe -k LocalService	Host Process for Wes
svchost.exe	928	00	4.888 K	C:\Windows\system32\svchost.exe -k NetworkService	Host Process for Wes
svchost.exe	948	00	10.888 K	C:\Windows\system32\svchost.exe -k netsvcs	Host Process for Wes
svchost.exe	1228	00	3.276 K	C:\Windows\system32\svchost.exe -k LocalServiceNoNetwork	Host Process for Wes
svchost.exe	1336	00	2.036 K	C:\Windows\system32\svchost.exe -k LocalServiceAndNoImpersonation	Host Process for Wes
svchost.exe	2684	00	1.732 K	C:\Windows\system32\svchost.exe -k imgsvc	Host Process for Wes
System	4	02	64 K		NT Kernel & Systen
System Idle Process	0	85	24 K		Percentage of time…
taskhost.exe	1872	00	2.776 K	"taskhost.exe"	Host Process for W
taskmgr.exe	1744	03	2.440 K	"C:\Windows\system32\taskmgr.exe" /1	Windows Task Man
UMVPFSrv.exe *32	980	00	524 K	"C:\Program Files (x86)\Common Files\logishrd\LVMVFM\UMVPFSrv.exe"	Logitech User mode
Unify.OpenScape.exe *32	2812	00	15.468 K	"C:\Program Files (x86)\Unify\OpenScape Desktop Client\Client\Unify.OpenScape.exe"	Unify.OpenScape.E
wininit.exe	424	00	216 K	wininit.exe	Windows Start-Up
winlogon.exe	516	00	272 K	winlogon.exe	Windows Logon Ap
wmpnetwk.exe	2508	00	3.256 K	"C:\Program Files\Windows Media Player\wmpnetwk.exe"	Windows Media Pla..

☑ Show processes from all users
End Process
Processes: 41 CPU Usage: 14% Physical Memory: 35%

Figure 24: Windows Task Manager (All Users)

- Via the Microsoft specific Tool (**Process Explorer**):

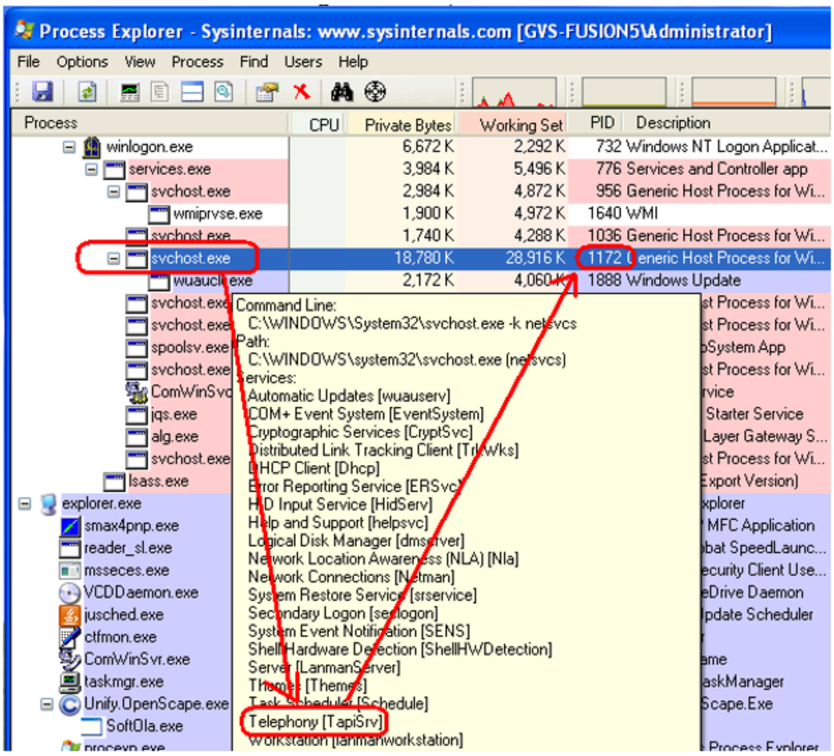
a) Start the Process Explorer:

   http://technet.microsoft.com/en-us/sysinternals/bb896653.aspx

b) Hover over **all svchost** processes.

   This action must be performed until the correct **PID** is identified.

   The correct **PID** is the one that has the telephony service **Telephony (TapiSrv)** under it. For example, the PID in the following figure is 1172.



Process Explorer - Sysinternals: www.sysinternals.com [GVS-FUSION5\Administrator]
File
Options
View
Process
Find
Users
Help

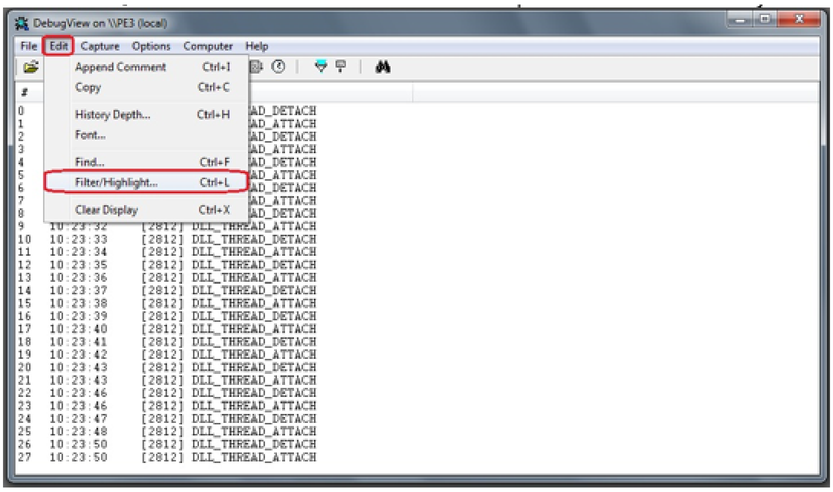
Process	CPU	Private Bytes	Working Set	PID	Description
winlogon.exe		6,672 K	2,292 K	732	Windows NT Logon Applicat...
services.exe		3,984 K	5,496 K	776	Services and Controller app
svchost.exe		2,984 K	4,872 K	956	Generic Host Process for Wi...
wmiprvse.exe		1,900 K	4,972 K	1640	WMI
svchost.exe		1,740 K	4,288 K	1036	Generic Host Process for Wi...
svchost.exe		18,780 K	28,916 K	1172	Generic Host Process for Wi...
wuauclt.exe		2,172 K	4,060 K	1888	Windows Update
svchost.exe					st Process for Wi...
svchost.exe					st Process for Wi...
spoolsv.exe					pSystem App
svchost.exe					st Process for Wi...
ComWinSvc					rvice
jqs.exe					Starter Service
alg.exe					Layer Gateway S...
svchost.exe					st Process for Wi...
lsass.exe					Export Version)
explorer.exe					xplorer
smax4pnp.exe					MFC Application
reader_sl.exe					bat SpeedLaunc...
msseces.exe					ecurity Client Use...
VCDDaemon.exe					eDrive Daemon
jusched.exe					pdate Scheduler
ctfmon.exe					
ComWinSvr.exe					ame
taskmgr.exe					askManager
Unify.OpenScape.exe					Scape.Exe
SoftOla.exe					
procexp.exe					Process Explorer

```
Command Line:
   C:\WINDOWS\System32\svchost.exe -k netsvcs
Path:
   C:\WINDOWS\system32\svchost.exe (netsvcs)
Services:
   Automatic Updates [wuauserv]
   COM+ Event System [EventSystem]
   Cryptographic Services [CryptSvc]
   Distributed Link Tracking Client [TrkWks]
   DHCP Client [Dhcp]
   Error Reporting Service [ERSvc]
   HID Input Service [HidServ]
   Help and Support [helpsvc]
   Logical Disk Manager [dmserver]
   Network Location Awareness (NLA) [Nla]
   Network Connections [Netman]
   System Restore Service [srservice]
   Secondary Logon [seclogon]
   System Event Notification [SENS]
   Shell Hardware Detection [ShellHWDetection]
   Server [LanmanServer]
   Themes [Themes]
   Task Scheduler [Schedule]
   Telephony [TapiSrv]
   Workstation [lanmanworkstation]
```

4) Select a **tool** to collect the traces, such **DbgViewor WinDbg**.

a) When using DbgView:

- Open DbgView as an administrator.
- Under the **Edit** menu, select **Filter/Highlight...**

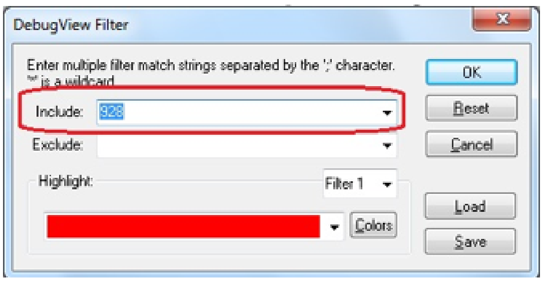


```
0                                   AD_DETACH
1                                   AD_ATTACH
2                                   AD_DETACH
3                                   AD_ATTACH
4                                   AD_DETACH
5                                   AD_ATTACH
6                                   AD_DETACH
7                                   AD_ATTACH
8                                   AD_DETACH
9    10:23:32    [2812]  DLL_THREAD_ATTACH
10   10:23:33    [2812]  DLL_THREAD_DETACH
11   10:23:34    [2812]  DLL_THREAD_ATTACH
12   10:23:35    [2812]  DLL_THREAD_DETACH
13   10:23:36    [2812]  DLL_THREAD_ATTACH
14   10:23:37    [2812]  DLL_THREAD_DETACH
15   10:23:38    [2812]  DLL_THREAD_ATTACH
16   10:23:39    [2812]  DLL_THREAD_DETACH
17   10:23:40    [2812]  DLL_THREAD_ATTACH
18   10:23:41    [2812]  DLL_THREAD_DETACH
19   10:23:42    [2812]  DLL_THREAD_ATTACH
20   10:23:43    [2812]  DLL_THREAD_DETACH
21   10:23:43    [2812]  DLL_THREAD_ATTACH
22   10:23:46    [2812]  DLL_THREAD_DETACH
23   10:23:46    [2812]  DLL_THREAD_ATTACH
24   10:23:47    [2812]  DLL_THREAD_DETACH
25   10:23:48    [2812]  DLL_THREAD_ATTACH
26   10:23:50    [2812]  DLL_THREAD_DETACH
27   10:23:50    [2812]  DLL_THREAD_ATTACH
```

Figure 25: DebugView – Edit menu

- Enter the **Process ID**.

  The example below displays the Process ID from the example presented above (**Process ID 928**).



Enter multiple filter match strings separated by the ';' character. '*' is a wildcard.

Include: 928

Exclude:

Highlight: Filter 1

Figure 26: Explorer Process ID

- Activate the **CaptureGlobalWin32** option.

> **NOTICE:** If possible, activate also **Capture Kernel**. This option might be disabled or it might be unavailable due to operating system policies.

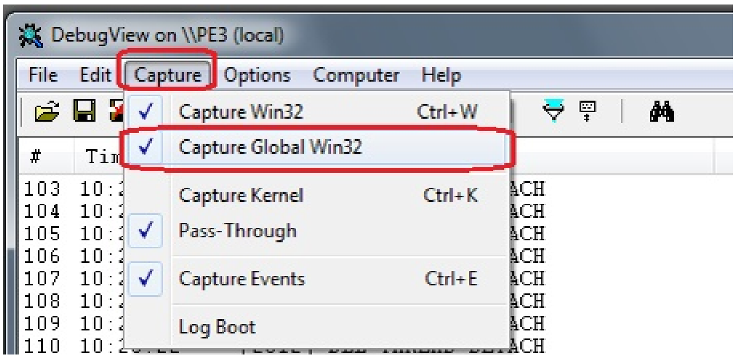


Figure 27: Activating "CaptureGlobalWin32" option

- If already connected (monitoring) to the **local PC**, the **PE3 (local)** option will be displayed inside the **Computer** menu:

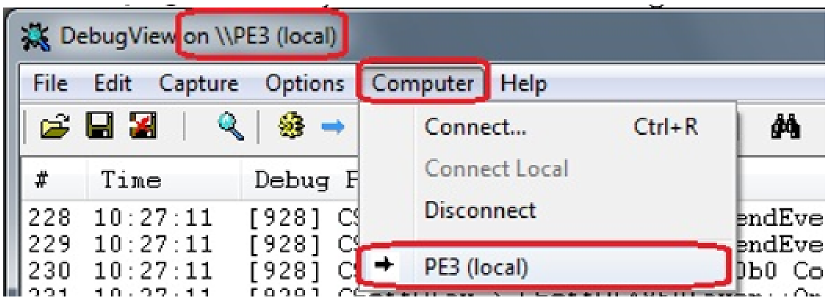

DebugView on \\PE3 (local)
File Edit Capture Options Computer Help
Connect... Ctrl+R
Connect Local
Disconnect
PE3 (local)

Figure 28: Connected to local PC option (example)

- If not already connected, then the option **Local PC** will be displayed inside the **Computer** menu. Click this option to connect to the local PC.

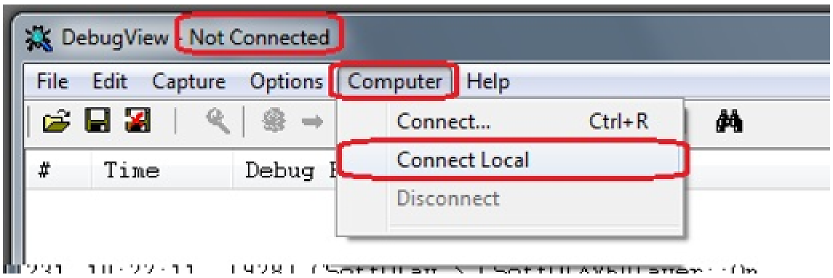
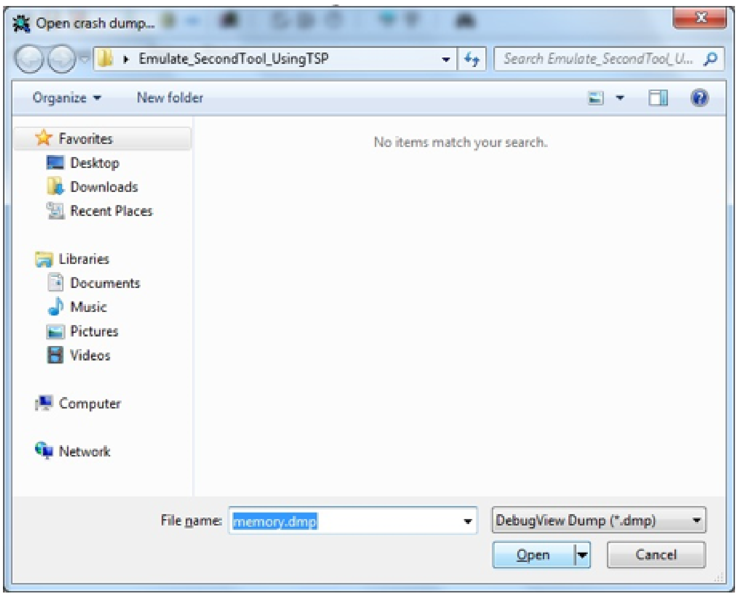

DebugView - Not Connected
File Edit Capture Options Computer Help
Connect... Ctrl+R
Connect Local
Disconnect

Figure 29: Select and connecting to local PC option (example)

- After performing the test, select a name for the log file and save it via the **File** menu, by selecting **Save** or **Save As**.
- Choose where to save the DUMP file via **ProcessCrashDump** option, displayed under the **File** menu.
- Select the folder where you want to store the DUMP file.



No items match your search.

File name: memory.dmp

DebugView Dump (*.dmp)

Figure 30: Selecting the Folder/file to save DUMP (example)

b) When using the **WinDbg** application:

- Open **WinDbg 64 bit** as an administrator.
- **Attach** the **process** to be monitored by **WinDbg**, by selecting the **Attach to a Process…** option, displayed under the **File** menu.

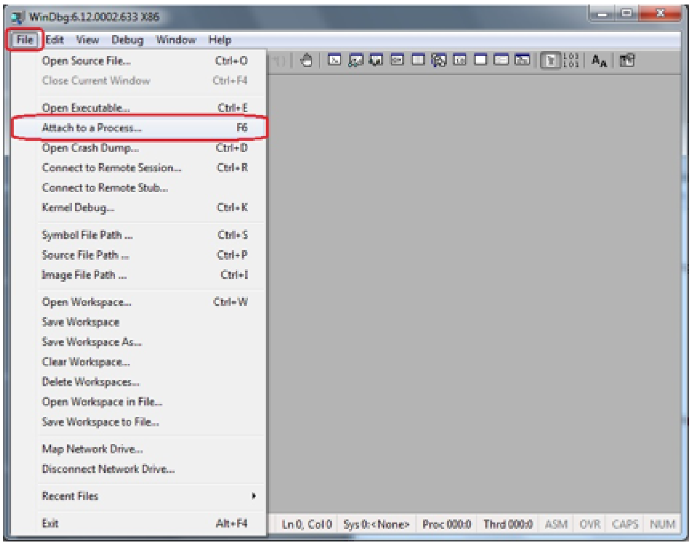
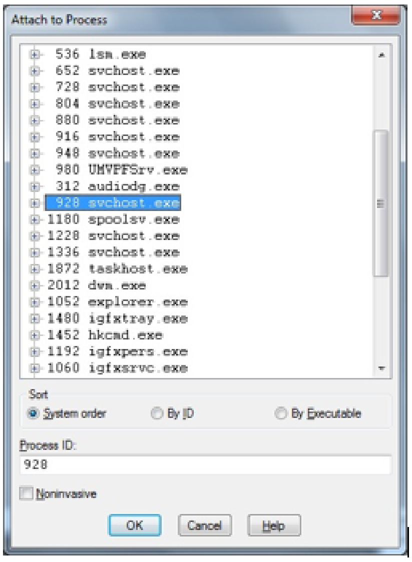

WinDbg:6.12.0002.633 X86
File
Edit
View
Debug
Window
Help
Open Source File... Ctrl+O
Close Current Window Ctrl+F4
Open Executable... Ctrl+E
Attach to a Process... F6
Open Crash Dump... Ctrl+D
Connect to Remote Session... Ctrl+R
Connect to Remote Stub...
Kernel Debug... Ctrl+K
Symbol File Path ... Ctrl+S
Source File Path ... Ctrl+P
Image File Path ... Ctrl+I
Open Workspace... Ctrl+W
Save Workspace
Save Workspace As...
Clear Workspace...
Delete Workspaces...
Open Workspace in File...
Save Workspace to File...
Map Network Drive...
Disconnect Network Drive...
Recent Files
Exit Alt+F4
Ln 0, Col 0
Sys 0:<None>
Proc 000:0
Thrd 000:0
ASM
OVR
CAPS
NUM

Figure 31: Attaching Process (example)

- Select the desired process you want to attach from the list of processes.



Attach to Process

```
536 lsm.exe
652 svchost.exe
728 svchost.exe
804 svchost.exe
880 svchost.exe
916 svchost.exe
948 svchost.exe
980 UMVPFSrv.exe
312 audiodg.exe
928 svchost.exe
1180 spoolsv.exe
1228 svchost.exe
1336 svchost.exe
1872 taskhost.exe
2012 dwm.exe
1052 explorer.exe
1480 igfxtray.exe
1452 hkcmd.exe
1192 igfxpers.exe
1060 igfxsrvc.exe
```

Sort

● System order ○ By ID ○ By Executable

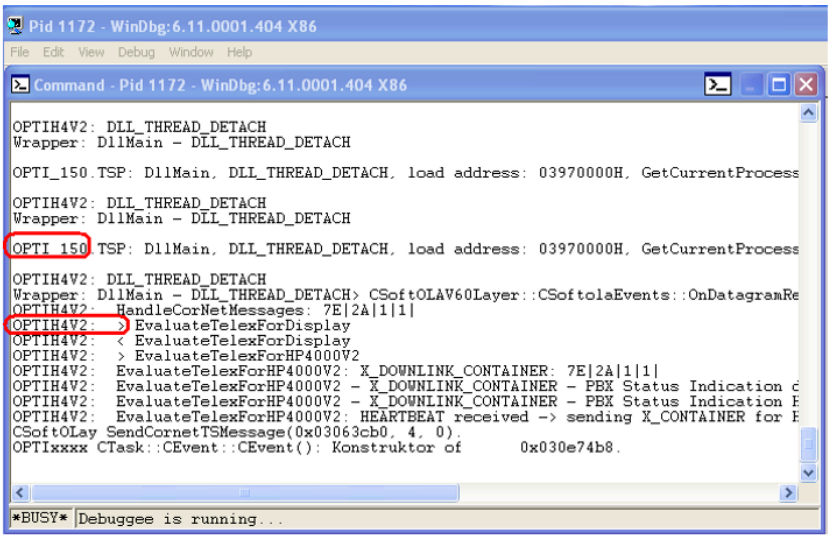
Process ID: 928

□ Noninvasive

Figure 32: Selecting the Process (example)

- Run the following command in the **WinDbg command line**: **sxe ***
- Run the following command in the **WinDbg command line**: **g**

The output of the above two commands should be similar to the one displayed in the figure below:



Pid 1172 - WinDbg:6.11.0001.404 X86
File Edit View Debug Window Help
Command - Pid 1172 - WinDbg:6.11.0001.404 X86

```
OPTIH4V2: DLL_THREAD_DETACH
Wrapper: DllMain - DLL_THREAD_DETACH

OPTI_150.TSP: DllMain, DLL_THREAD_DETACH, load address: 03970000H, GetCurrentProcess

OPTIH4V2: DLL_THREAD_DETACH
Wrapper: DllMain - DLL_THREAD_DETACH

OPTI_150.TSP: DllMain, DLL_THREAD_DETACH, load address: 03970000H, GetCurrentProcess

OPTIH4V2: DLL_THREAD_DETACH
Wrapper: DllMain - DLL_THREAD_DETACH> CSoftOLAV60Layer::CSoftolaEvents::OnDatagramRe
OPTIH4V2:  HandleCorNetMessages: 7E|2A|1|1|
OPTIH4V2:  > EvaluateTelexForDisplay
OPTIH4V2:  < EvaluateTelexForDisplay
OPTIH4V2:  > EvaluateTelexForHP4000V2
OPTIH4V2:  EvaluateTelexForHP4000V2: X_DOWNLINK_CONTAINER: 7E|2A|1|1|
OPTIH4V2:  EvaluateTelexForHP4000V2 - X_DOWNLINK_CONTAINER - PBX Status Indication c
OPTIH4V2:  EvaluateTelexForHP4000V2 - X_DOWNLINK_CONTAINER - PBX Status Indication F
OPTIH4V2:  EvaluateTelexForHP4000V2: HEARTBEAT received -> sending X_CONTAINER for F
CSoftOLay SendCornetTSMessage(0x03063cb0, 4, 0).
OPTIxxxx CTask::CEvent::CEvent(): Konstruktor of      0x030e74b8.
```

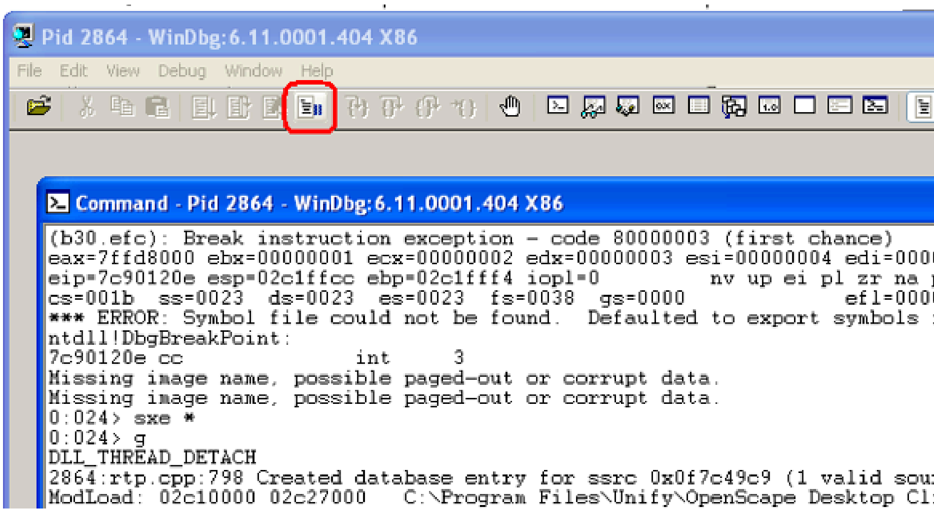
BUSY Debuggee is running...

Figure 33: WinDbg Debug running window (example)

In case of HP4K traces, there will be many lines with the **OPTIH4V2** header.

In case of H3K or OpenScape Office traces, there will be many lines with the **OPTI_150** header.

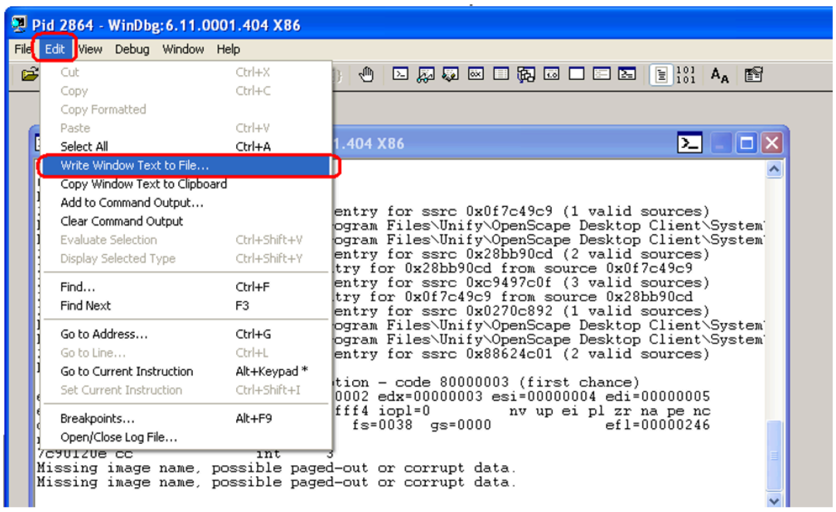
- Perform the test, then break the process.


Pid 2864 - WinDbg:6.11.0001.404 X86
File Edit View Debug Window Help
Command - Pid 2864 - WinDbg:6.11.0001.404 X86

```
(b30.efc): Break instruction exception - code 80000003 (first chance)
eax=7ffd8000 ebx=00000001 ecx=00000002 edx=00000003 esi=00000004 edi=000
eip=7c90120e esp=02c1ffcc ebp=02c1fff4 iopl=0         nv up ei pl zr na p
cs=001b  ss=0023  ds=0023  es=0023  fs=0038  gs=0000             efl=000
*** ERROR: Symbol file could not be found.  Defaulted to export symbols
ntdll!DbgBreakPoint:
7c90120e cc              int     3
Missing image name, possible paged-out or corrupt data.
Missing image name, possible paged-out or corrupt data.
0:024> sxe *
0:024> g
DLL_THREAD_DETACH
2864:rtp.cpp:798 Created database entry for ssrc 0x0f7c49c9 (1 valid sou
ModLoad: 02c10000 02c27000   C:\Program Files\Unify\OpenScape Desktop Cl
```

Figure 34: WinDbg – Breaking process AFTER tests performed (example)

- Save the trace by selecting the **Write Window Text to File…** option from the **Edit** menu.


Pid 2864 - WinDbg:6.11.0001.404 X86
File Edit View Debug Window Help
Cut Ctrl+X
Copy Ctrl+C
Copy Formatted
Paste Ctrl+V
Select All Ctrl+A
Write Window Text to File…
Copy Window Text to Clipboard
Add to Command Output…
Clear Command Output
Evaluate Selection Ctrl+Shift+V
Display Selected Type Ctrl+Shift+Y
Find… Ctrl+F
Find Next F3
Go to Address… Ctrl+G
Go to Line… Ctrl+L
Go to Current Instruction Alt+Keypad *
Set Current Instruction Ctrl+Shift+I
Breakpoints… Alt+F9
Open/Close Log File…
1.404 X86

```
entry for ssrc 0x0f7c49c9 (1 valid sources)
ogram Files\Unify\OpenScape Desktop Client\System
ogram Files\Unify\OpenScape Desktop Client\System
entry for ssrc 0x28bb90cd (2 valid sources)
try for 0x28bb90cd from source 0x0f7c49c9
entry for ssrc 0xc9497c0f (3 valid sources)
try for 0x0f7c49c9 from source 0x28bb90cd
entry for ssrc 0x0270c892 (1 valid sources)
ogram Files\Unify\OpenScape Desktop Client\System
ogram Files\Unify\OpenScape Desktop Client\System
entry for ssrc 0x88624c01 (2 valid sources)

tion - code 80000003 (first chance)
0002 edx=00000003 esi=00000004 edi=00000005
fff4 iopl=0         nv up ei pl zr na pe nc
  fs=0038  gs=0000             efl=00000246
7c90120e cc              int     3
Missing image name, possible paged-out or corrupt data.
Missing image name, possible paged-out or corrupt data.
```

Figure 35: WinDbg – Saving log (example)

- Finally, detach the process by selecting the **Detach Debuggee** option from the **Debug** menu.

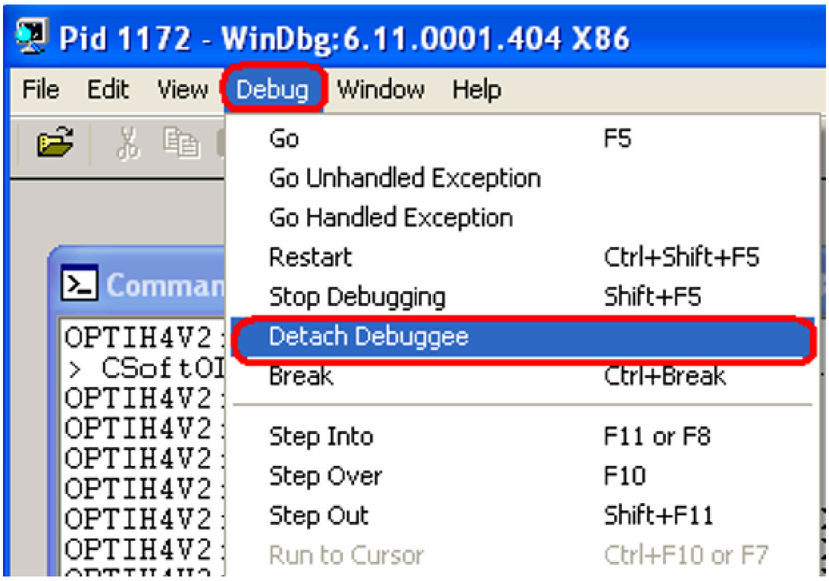


Figure 36:

NOTICE: This action is very important, otherwise the system will be unstable, because the **svchost** process has several other services under it.

3.11.4 Annex D (Performance Monitor)

This section describes how to collect logs using **MS Performance Monitor**. These logs might be requested by Microsoft or the Dev team to further analyze the behaviour of the operating system or the status of a specific tool.

Follow the steps below to generate a log via **MS Performance Monitor**:

1) Click the Windows **Start** button, type `perfmon` and click `perfmon.exe`.
2) Create a new **Data Collector Set** by navigating to **Data Collector Sets \User Defined** in the left hand panel, right clicking on **User Defined**, then selecting **New\Data Collector Set** from the context menu.

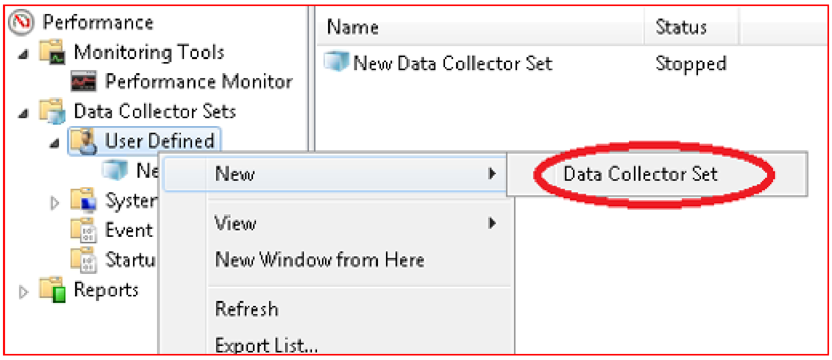
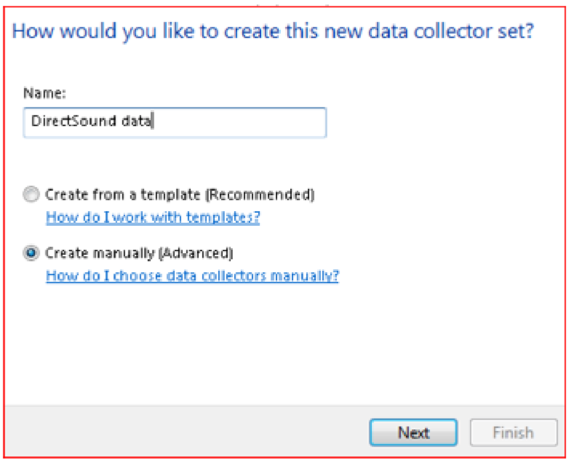


Figure 37: Perfmon – Creating new Data Collector Set

3) Enter a name for the **Data Collector Set** in the **Name** field and select the **Create manually (Advanced)**. Finally, click **Next**.



How would you like to create this new data collector set?

Name: DirectSound data

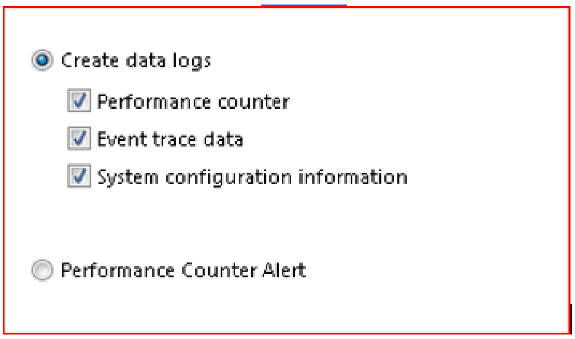
○ Create from a template (Recommended)

● Create manually (Advanced)

How do I work with templates?
How do I choose data collectors manually?
Next
Finish

Figure 38: Perfmon – entering a Name

4) Click **Create data logs**, then enable the **Performance counter, Event trace data and System configuration information** options and click **Next** .



● Create data logs

☑ Performance counter

☑ Event trace data

☑ System configuration information

○ Performance Counter Alert

Figure 39: Perfmon – Selecting Data logs types

5) Click **Add**.

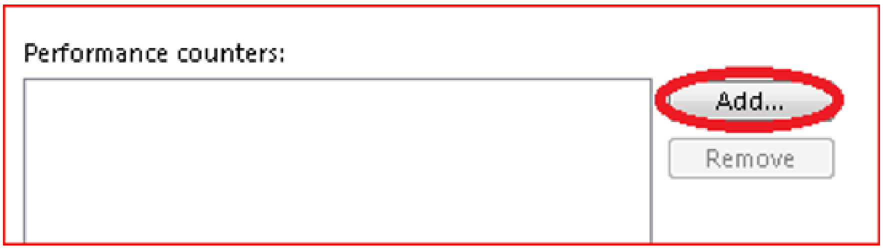
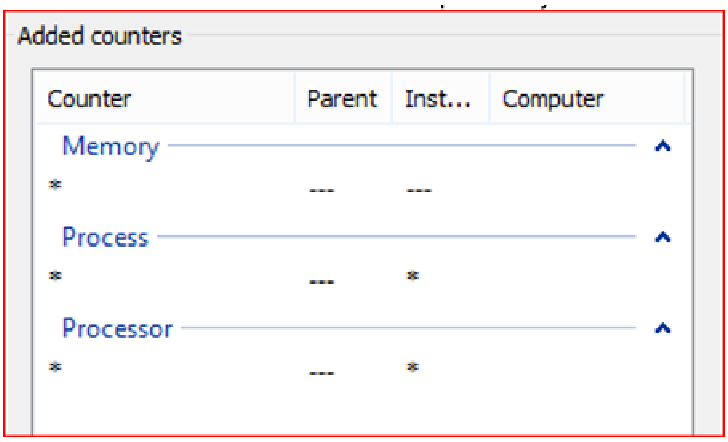


Figure 40: Perfmon – Adding performance counters stepA

6) Add the following counters:

- **Process** - <All instances>
- **Processor** - <All instances>
- **Memory**

In the **Added counters** panel, you can view the counters added:



Added counters

Counter	Parent	Inst...	Computer
Memory			
*	---	---	
Process			
*	---	*	
Processor			
*	---	*	

Figure 41: Perfmon – Adding performance counters stepB

7) Click **OK** and **Finish** to create the data set.

8) Before running the data set, right click on it, select **Properties**, and set the desired output location on the **Directory**, and click **OK**.

9) You can start or stop the logs by selecting the corresponding data set and the clicking the play/stop **buttons** on the menu bar.

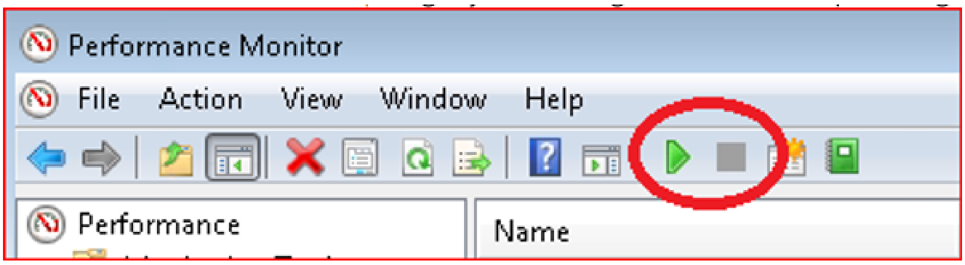


Figure 42: Perfmon – Start/Stop logs