Бесплатная электронная книга

УЧУСЬ

WinDbg

Free unaffiliated eBook created from
Stack Overflow contributors.

#windbg

1

1: WinDbg 2

2

2

Examples 3

3

Debuggers 3

2: DML ( ) 4

Examples 4

/ 4

3: 5

Examples 5

5

4: / 6

Examples 6

6

6

6

7

7

, , 7

8

8

8

9

WinDbg 9

9

9

Windbg 10

5: 12

Examples 12

12

6: **13**

Examples 13

13

SOSex 13

PyKD 13

PyKd 13

NetExt 14

14

CoSOS 15

7: **16**

Examples 16

16

17

Около

You can share this PDF with anyone you feel could benefit from it, downloaded the latest version from: windbg

It is an unofficial and free WinDbg ebook created for educational purposes. All the content is extracted from Stack Overflow Documentation, which is written by many hardworking individuals at Stack Overflow. It is neither affiliated with Stack Overflow nor official WinDbg.

Use the content presented in this book at your own risk; it is not guaranteed to be correct nor accurate, please send your feedback and corrections to info@zzzprojects.com

глава 1: Начало работы с WinDbg

замечания

В этом разделе представлен обзор того, что такое windbg, и почему разработчик может захотеть его использовать.

Следует также упомянуть любые крупные темы в windbg и ссылки на связанные темы. Поскольку Documentation for windbg является новым, вам может потребоваться создать начальные версии этих связанных тем.

Версии

Важные версии WinDbg для поддерживаемых версий WinDbg. См. Также подробный список с историческими версиями онлайн.

Важно отметить, что с версией 6.12 изменилась схема версий версий до более новой версии 6.1. Более старые версии имеют низкое число (<100) на третьем месте, а более новые версии - с большими номерами (> 6000).

Во многих случаях версии WinDbg для новых версий Windows по-прежнему работают в более старых версиях в Windows, например, версия 10 WinDbg все еще может использоваться в Windows 7. Однако некоторые команды могут использовать вызовы API, которые недоступны и, таким образом, не работают. Поэтому хорошо иметь несколько версий WinDbg.

Версия	Описание	Дата выхода
6.12.0002.633	для Windows 7 и .NET Framework 4	2010-05-21
6.1.7600.16385		2009-07-24
6.2.8400.0	обновление для Windows 8 (?)	2012-06-23
6.2.9200.16384	для Windows 8 и .NET Framework 4.5	2012-11-15
6.3.9600.16384	для Windows 8.1	2013-10-17
10.0.10075.9	для Windows 10	2015-04-29
10.0.10586.567	предоставляется с Windows 10, постройте 1511	2015-10-30
10.0.14321.1024	предоставляется с Windows 10, постройте 1607	2016-07-29

Examples

Установка или настройка

Microsoft описывает три способа установки WinDbg:

- как часть WDK (комплект драйверов Windows)
- как часть SDK (Software Development Kit)
- с установщиком SDK и снятием выделения с остального, кроме «Отладки для Windows»,

Чтобы получить программу установки, посетите страницу загрузки WDK, WinDbg и связанных с ней инструментов и прокрутите вниз до раздела «Получить инструменты отладки».

Известным и удобным, но неофициальным источником является Codemachine, где вы также можете напрямую загрузить более старые версии средств отладки.

Сама установка прямолинейна. Пройдите через установщик, пока он не закончится.

Debuggers

WinDbg часто используется как аббревиатура «Средства отладки для Windows». Он содержит различные отладчики:

дебаггер	Описание
WinDbg	отладчик с графическим интерфейсом пользователя
CDB	**c** onsole **d** e **b** ugger, отладчик пользовательского режима, который запускается в текущей открытой консоли
НЦД	**n** ew **t** erninal **s** ymbolic **d** ebugger, отладчик пользовательского режима, который открывает новый терминал (консоль), как следует из названия
KD	**k** ernel **d** ebugger, который работает в открытой консоли
NTKD	**n** ew **t** erninal **k** ernel **d** ebugger, открывает новый терминал

Команды идентичны, за исключением того, что могут быть команды, связанные с GUI, которые не работают в консольных версиях.

Прочитайте Начало работы с WinDbg онлайн: https://riptutorial.com/ru/windbg/topic/1833/начало-работы-с-windbg

глава 2: DML (язык отладки отбраковки)

Examples

Включение / выключение

.prefer_dml 1 включить вывод dmlformat

.prefer_dml 0 отключить вывод dmlformat

Прочитайте DML (язык отладки отбраковки) онлайн:
https://riptutorial.com/ru/windbg/topic/7987/dml--язык-отладки-отбраковки-

глава 3: Анализ сбоев

Examples

Анализ сбоев в базовом пользовательском режиме

`.exr -1` дает вам сведения о последнем исключенном исключении.

`!analyze -v` обычно хорошо работает.

Для .NET команда `!pe` расширения SOS показывает подробности об исключении .NET, которое было выбрано.

Прочитайте Анализ сбоев онлайн: https://riptutorial.com/ru/windbg/topic/5389/анализ-сбоев

глава 4: Отладка пользовательского режима / приложения

Examples

Важные команды

Документирование вашей работы

Помните, что вы сделали, и сохраните длинные выходы, которые нельзя сохранить в буфере WinDbg. Всегда полезно иметь журнал для воспроизведения шагов отладки, например, задавать вопросы о переполнении стека.

команда	Цель
`.logopen`	создать файл журнала
`.logclose`	закрыть файл журнала
`.dump`	save crash dump file (моментальный снимок текущей сессии отладки)

Работа с символами

Без или с неправильными символами вы можете получить неверную информацию и ввести в заблуждение. Перед началом работы в WinDbg убедитесь, что вы знакомы с этими командами. См. Также Как настроить символы в WinDbg .

команда	Цель
`.symfix`	установить или добавить символы в официальный путь символа Microsoft
`.sympath`	устанавливать или добавлять собственные или сторонние символы
`.reload`	перезагрузить символы
`.symopt`	определение параметров обработки символов
`!sym`	загрузка управляющего символа
`x`	проверять символы
`ln`	список ближайших символов

Анализ сбоев

Узнайте, что произошло (в аварийных дампах) и как обрабатывать события (в режиме живой отладки).

команда	Цель
`.exr`	отображать запись об исключении
`.lastevent`	отображать последнее событие
`sx`	определять обработку исключений
`!analyze`	анализировать крушение или висеть
`!avrf`	верификатор приложения

Окружающая среда

Проверьте имя процесса и информацию о версии.

команда	Цель
| (Труба)	обрабатывать информацию
`lm`	список модулей

Темы, стеки вызовов, регистры и память

Проверьте детали.

команда	Цель
`~`	список тем
`r`	регистры
`k`	стек вызовов
`d` *	отображать память
`e` *	редактировать память
`s`	память поиска

команда	Цель
`.formats`	конвертировать между форматами чисел
`?`	оценивать выражение
`u *`	разбирать
`a`	собирать
`!address`	информация о памяти

Управление целевым

В живой отладке возьмите управление исполнением.

команда	Цель
`g`	go / continue
`gu`	подниматься
`p`	Единственный шаг
`t`	трассировка (одношаговые и выходные регистры)
`bp`	установить точку останова
`bl`	список контрольных точек

Работа с расширениями

Расширения могут обеспечить значительные преимущества и улучшения.

команда	Цель
`.load`	расширение нагрузки (полный путь)
`.loadby`	расширение нагрузки относительно модуля
`.chain`	отображать загруженные расширения
`.unload`	выгружать расширение

Остановить отладку

команда	Цель
`q`	прекратить и закрыть приложение
`qd`	отделить и выйти

Прикрепите и отсоедините

команда	Цель
`.tlist`	список процессов
`.attach`	приложить к процессу
`.create`	создать процесс и приложить
`.childdbg`	определить поведение отладки дочернего процесса
`.detach`	отрываться от процесса
`.kill`	убить процесс
`.restart`	перезапустить процесс

Поведение WinDbg

команда	Цель
`.prefer_dml`	установить обработку языка разметки отладчика
`.effmach`	переключить бит

Команды юзабилити

команда	Цель
`.cmdtree`	Загружает текстовый файл с предопределенными командами в отдельном окне

Получение помощи

команда	Цель
`.hh`	Отображает справочное руководство для команд WinDbg

Создание пользовательского окна команд в Windbg

Команда `.cmdtree` позволяет открыть файл `.txt` с предопределенными командами, которые вы можете просто дважды щелкнуть, чтобы выполнить.

Как создать файл команды

Создайте файл с помощью этого шаблона

```
windbg ANSI Command Tree 1.0
title {"Window title"}
body
{"Group Heading"}
 {"Name of command to display"} {"command"}
 {"Name of command to display"} {"command"}
{"Group Heading"}
 {"Name of command to display"} {"command"}
```

Что нужно позаботиться

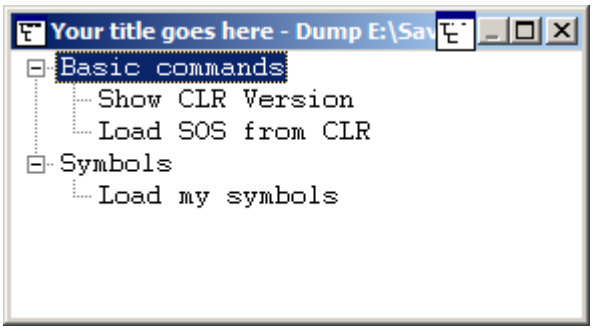
1. Формат шаблона должен соблюдаться именно для открытия файла в Windbg.
2. Новая строка требуется после каждого `{Group Heading}` .
3. Каждая пара `{Name of command to display} {command}` должна быть в одной строке и должна сопровождаться новой строкой.

Пример пользовательского командного файла

```
windbg ANSI Command Tree 1.0
title {"Your title goes here"}
body
{"Basic commands"}
 {"Show CLR Version"} {"lmv m clr"}
 {"Load SOS from CLR"} {".loadby sos clr "}
{"Symbols"}
 {"Load my symbols"} {".sympath+ "c:\DebugSymbols" ; .reload"}
```

Как открыть командный интерфейс из окна команд

Выполните `.cmdtree <path of your .txt file>` чтобы открыть окно. Вы увидите окно, подобное этому



Дважды щелкните команду для выполнения.

Прочитайте Отладка пользовательского режима / приложения онлайн:
https://riptutorial.com/ru/windbg/topic/5384/отладка-пользовательского-режима---приложения

глава 5: Отладка ядра

Examples

Важные команды

- ! процесс - список процессов пользовательского режима
- .process - установить контекст процесса
- ! peb-show process environment block
- ! teb - показать поток среды
- ! блокировки - анализ взаимоблокировки
- .dump - сохранить файл аварийного дампа на диск

Прочитайте Отладка ядра онлайн: https://riptutorial.com/ru/windbg/topic/6076/отладка-ядра

глава 6: расширения

Examples

СОС

SOS (сын забастовки) является официальным расширением WinDbg от Microsoft для .NET. Он устанавливается как часть платформы .NET и, таким образом, доступен по умолчанию.

Как и любое расширение, его можно загрузить с помощью `.load x:\full\path\to\sos.dll` , но есть более простые способы. В зависимости от версии .NET расширение расположено рядом с `mscorwks.dll` (.NET CLR 2), `clr.dll` (.NET CLR 4) или `coreclr.dll` (Silverlight и Universal apps), поэтому один из следующие команды должны работать:

```
.loadby sos clr
.loadby sos coreclr
.loadby sos mscorwks
```

Список доступных команд см. `!help` .

SOSex

SOSex является расширением SOS, написанным Стивом Джонсоном , сотрудником Microsoft. Он бесплатно предоставляет SOSex для скачивания , но это не с открытым исходным кодом.

Как правило, расширение не доступно рядом с любой другой DLL, поэтому обычно загружается с `.load x:\full\path\to\sosex.dll` .

Помимо упрощения отладки .NET, команда `!dlk` также может использоваться в собственных средах для проверки тупиков критических разделов.

Для получения списка доступных команд, обратитесь `!help` в SOSex.

PyKD

PyKD - это расширение WinDbg, которое позволяет писать сценарии Python. Это с открытым исходным кодом.

Как правило, расширение не доступно рядом с любой другой DLL, поэтому обычно оно загружается с помощью `.load x:\full\path\to\pykd.pyd` , где PYD является расширением для библиотеки python, но вы можете переименовать это в DLL, если хотите.

Начало работы с PyKd

PyKD не предлагает `!help` Help, поэтому посмотрите документацию на Codeplex. Многие разработчики, похоже, из России, и самая современная и полная документация, вероятно, на русском языке. Переводчик Google выполняет достойную работу.

Как и другие расширения, используйте правильную битту расширения, которая соответствует версии WinDbg. Кроме того, у вас должен быть установлен Python с той же самой битностью.

`!py` запускает REPL-интерпретатор и `!py x:\path\to\script.py` запускает скрипт python. Скрипты должны использовать

```
from pykd import *
```

как первая строка, чтобы использовать функциональность PyKD, в то время как эта строка не нужна в интерпретаторе REPL. С помощью `exit()` можно вывести интерпретатор.

NetExt

NetExt - это расширение для .NET, которое обеспечивает

- LINQ-подобные запросы для объектов в куче (`!wselect !wfrom`)
- возможности отображения для специальных объектов, таких как словари и хеш-таблицы (`!wdict !whash`)
- Команды, связанные с ASP.NET / HTTP (`!wcookie !wruntime !whttp`)
- несколько других связанных с сетью команд

Как правило, расширение не доступно рядом с любой другой DLL, поэтому обычно загружается с .load x: \ full \ path \ to \ netext.dll

Обзор расширений

Неполный список расширений WinDbg, которые не установлены с самим WinDbg:

расширение	Цель
COC	.NET (официальное расширение Microsoft)
SOSex	.NET (расширение для SOS)
CoSOS	.NET (расширение для SOS)
NetExt	.NET (с фокусом на сети)
PyKD	Пиратские скрипты
PDE	Служебные приложения Windows и хранилища (убранные исключения)

расширение	Цель
PSSCOR	.СЕТЬ
SDbgExt	.СЕТЬ
MEX	.СЕТЬ

CoSOS

CoSOS (кузен SOS) является расширение с открытым исходным кодом для WinDbg сосредоточив внимание на фрагментации .NET памяти (`!gcview`) и нарезание резьб вопросы (`!wfo` , `!tn`).

Как правило, расширение не доступно рядом с любой другой DLL, поэтому обычно загружается с `.load x:\full\path\to\cosos.dll` . Для этого требуется, чтобы SOS был загружен и в настоящее время работает только с 32-разрядными приложениями.

Прочитайте расширения онлайн: https://riptutorial.com/ru/windbg/topic/5391/расширения

глава 7: Удаленная отладка

Examples

Важные команды

- .server - создать сервер отладки
- .clients - список отладочных клиентов, подключенных к серверу.
- .endsrv - завершение работы сервера отладки
- .servers - список подключений к серверу отладки
- .remote - запустите сервер remote.exe
- .noshell - предотвращать команды оболочки

Прочитайте Удаленная отладка онлайн: https://riptutorial.com/ru/windbg/topic/5977/удаленная-отладка

кредиты

S. No	Главы	Contributors
1	Начало работы с WinDbg	Community, Thomas Weller
2	DML (язык отладки отбраковки)	Wang Zhengzhang
3	Анализ сбоев	Thomas Weller
4	Отладка пользовательского режима / приложения	Piyush Parashar, Thomas Weller, X. Liu
5	Отладка ядра	Thomas Weller
6	расширения	Jason Evans, Lieven Keersmaekers, Thomas Weller
7	Удаленная отладка	Thomas Weller