|  |
| --- |
|  |
| Ollydbg |
| Petar Afrić, 0036469979 |

|  |
| --- |
| Mentor: prof. Marin Golub  Akademska godina 2014/2015 |

# Sadržaj

[0 Sadržaj 1](#_Toc405926128)

[1 Uvod 2](#_Toc405926129)

[1.1 Sistemski zahtjevi za Ollydbg 2](#_Toc405926130)

[1.2 Privatnost korisnika Ollydbg-a 3](#_Toc405926131)

[1.3 Preuzimanje Ollydbg programa 3](#_Toc405926132)

[2 Grafičko sučelje 4](#_Toc405926133)

[2.1 Meni 4](#_Toc405926134)

[2.2 Glavna alatna traka 5](#_Toc405926135)

[2.3 Glavni prozor 6](#_Toc405926136)

[2.3.1 CPU/kodni prozor 7](#_Toc405926137)

[2.3.2 Registarski prozor 8](#_Toc405926138)

[2.3.3 Prozor stoga 8](#_Toc405926139)

[2.3.4 Prozor memorijskog spremišta 9](#_Toc405926140)

[2.4 Statusna traka 10](#_Toc405926141)

[3 Osnove korištenja Ollydbg alata 11](#_Toc405926142)

[3.1 Traženje tekstualnih nizova 11](#_Toc405926143)

[3.2 Manipulacija koda 12](#_Toc405926144)

[3.3 Postavljanje prekidnih točaka 13](#_Toc405926145)

[4 Ollydbg i analiza zloćudnih programa 15](#_Toc405926146)

[5 Primjeri analize - Pristup ograničenom dijelu programa 16](#_Toc405926147)

[6 Zaključak 22](#_Toc405926148)

[7 Literatura 23](#_Toc405926149)

# Uvod

Ollydbg je 32 bitni debugger asemblerske razine za Microsoft Windows. Ollydbg daje poseban naglasak na binarnu analizu koda što ga čini izuzetno korisnim u slučajevima analize kada izvorni kod programa nije dostupan. Ollydbg omogućuje praćenje stanja registara, prepoznavanje procedura, poziva aplikacijskog programskog sučelja, prekidača, tablica, konstanti i znakovnih nizova. Program je shareware, što znači da iako je besplatan za korištenje potrebno je registriranje kako bi se omogućilo njegovo korištenje. Glavne prednosti Ollydbg-a su intuitivno shvatljivo sučelje i mogućnost nadogradnje od treće stranke preko plugin-a, dok mu je glavna mana nemogućnost analize programa compile-iranja za 64-bitne procesore.

Ollydbg nudi mnoge mogućnosti odnosno funkcionalnosti pri analizi strojnog koda programa. Neke od tih funkcionalnosti su:

* Direktno učitavanje i debuggiranje DDL-a
* Skeniranje datoteka objekata- lociranje i prepoznavanje objekata i kompletnih objektnih knjižnica
* Omogućuje korisničko definiranje labela, komentara i opisa funkcija
* Razumijevanje debbug informacija zapisanih u Borland formatu
* Debugg-iranje višedretvenih aplikacija tj. programa
* Analiza programa koji se trenutno izvode
* Dinamičko prepoznavanje Ascii i Unicode tekstualnih zapisa
* Prepoznavanje zahtjevnih programskih struktura kao što je poziv metode / funkcije
* Omogućuje postavljanje uvjetnih, logging, memorijskih i hardware-skih prekidnih točaka
* Omogućuje praćenje izvođenja programa te uvid u trenutna stanja varijabli u programu
* Omogućuje praćenje sadržaja sistemskog stoga
* Omogućuje pretraživanje cjelokupne programom rezervirane tj. alocirane memorije
* Omogućuje izmjene stanja memorije tijekom izvođenja programa
* Omogućuje zaustavljanje izvođenja programa u proizvoljnom vremenskom trenutku

## Sistemski zahtjevi za Ollydbg

Za pokretanje Ollydbg-a potrebno je imati sustav koji zadovoljava minimalno iduće zahtjeve:

* 586/100 procesor
* Microsoft Windows 95, Microsoft Windows 98, Microsoft Windows NT 4.0 ,Microsoft Windows 2000, Microsoft Windows Vista, Microsoft Windows 7 ili Microsoft Windows 8
* 64 MB fizičke memorije iako je za razumno funkcioniranje sustava barem 128 MB preporučeno
* 2MB slobodnog diskovnog prostora

Za analizu programa na operacijskom sustavu Microsoft Windows 2000 i svim njegovim sljedbenicima potrebno je program pokrenuti s administrativnim ovlastima kako bi ispravno funkcionirao.

## Privatnost korisnika Ollydbg-a

Iako Oleh Yuschuk, autor Ollydbg-a garantira da originalna verzija :

* Nikad ne analizira ni jedan proces osim onog koji se ciljano analizira.
* Nikad se ne ponaša kao mrežni klijent ili server
* Nikad ne šalje nikakve podatke nekom drugom sustavu
* Nikad se ne ponaša kao trojanski konj
* Nikad ne čita sistemske registre osim ukoliko se to eksplicitno od njega ne traži
* Nikad ne mijenja sadržaj sistemskih registara osim ukoliko se to eksplicitno od njega ne traži
* Mijenjanje sistemskih registara je ograničeno na :
  + HKEY\_CLASSES\_ROOT\exefile\shell\Open with Ollydbg
  + HKEY\_CLASSES\_ROOT\exefile\shell\Open with Ollydbg\command
  + HKEY\_CLASSES\_ROOT\dllfile\shell\Open with Ollydbg
  + HKEY\_CLASSES\_ROOT\ dllfile \shell\Open with Ollydbg\command
  + HKEY\_LOCAL\_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrnetVersion\AeDebug\Debugger
  + HKEY\_LOCAL\_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrnetVersion\AeDebug\Auto
* Ne stvara, mijenja ili briše sistemske datoteke
* Nikad ne mijenja izvršne datoteke ili DLL osim ukoliko se to eksplicitno od njega ne traži
* Ne pamti informacije o provedenim analizama
* Ne stvara i mijenja datoteke izvan direktorija u kojem se nalazi

Zbog njegove česte uporabe pri analizi malicioznog odnosno zloćudnog koda i lake mogućnosti nadogradnje prilikom preuzimanja programa preporuča se pregled preuzetih datoteka s antivirusnim programima. Nerijetko se događa da treće strane zlonamjernih namjera dodaju svoje funkcionalnosti koje na prvi pogled imaju korisnu funkciju, ali u pozadini obavljaju zloćudne aktivnosti. Najčešće su napadi usmjereni na pamćenje provedenih analiza te slanje informacija o njima napadaču.

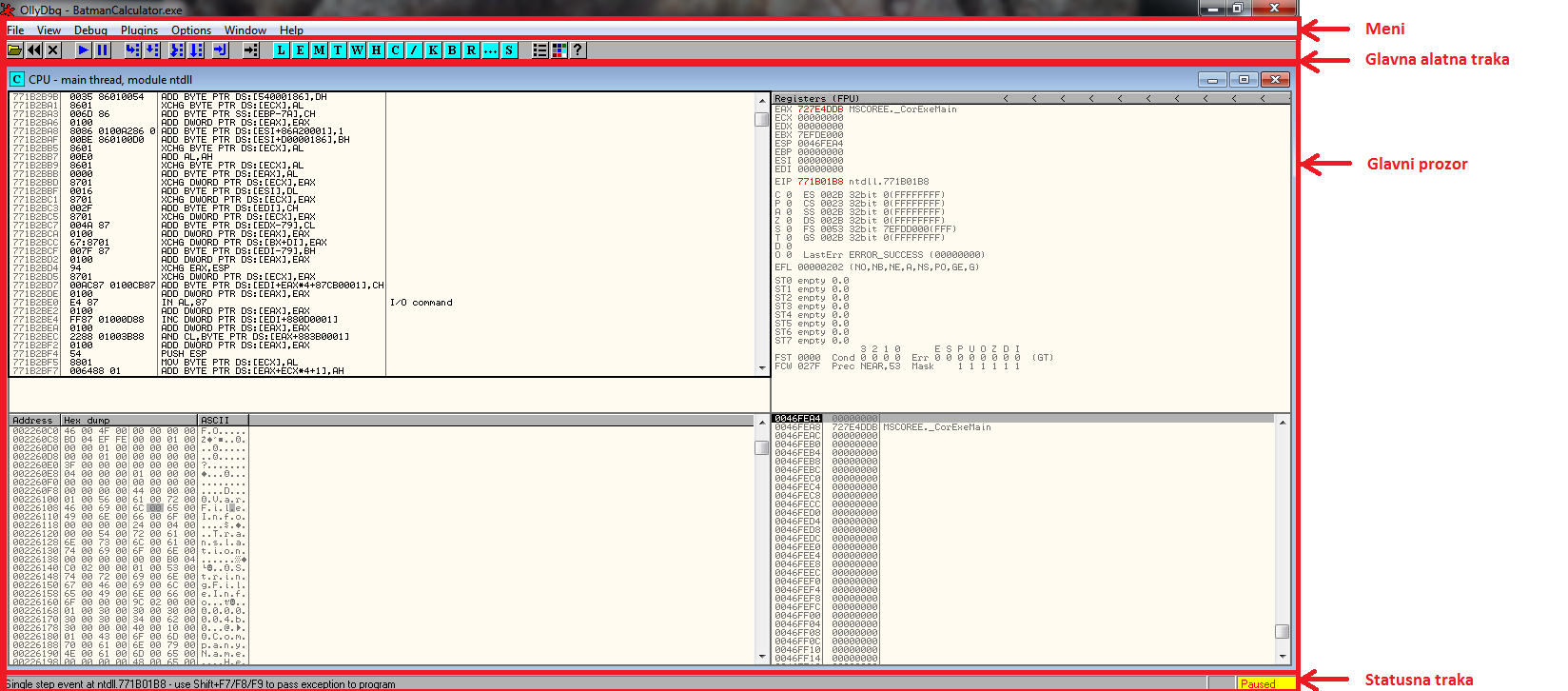
## Preuzimanje Ollydbg programa

Preuzimanje programa Ollydbg moguće je izvesti na stranici: <http://www.ollydbg.de/download.htm> . Preuzimanje je moguće izvesti i bez registriranja ali se za korištenje programa u komercijalne svrhe te na dulji vremenski period preporuča registriranje. Za registriranje je potrebno poslati mail s ispunjenim registracijskim formularom na mail adresu: [Ollydbg@t-online.de](mailto:Ollydbg@t-online.de) . Važno je naglasiti da su sve osobne informacije u spomenutom formularu sasvim opcionalno. Cilj registracije je više dati autoru Ollydbg-a uvid u broj korisnika njegove programske potpore nego bilo što drugo. Prilikom registracije moguće je odabrati opciju za dobivanje informacija o programu i njegovom razvoju. Autor Ollydbg-a garantira da broj informacijskih mail-ova neće prelaziti četiri mail-a godišnje. Jednom preuzet program nije potrebno instalirati već samo raspakirati u neki direktorij.

# Grafičko sučelje

Grafičko sučelje programa Ollydbg je izuzetno intuitivno i lako za shvaćanje. Grafičko sučelje moguće je podijeliti na četiri dijela:

* Meni – glavni izbornik
* Glavna alatna traka
* Glavni prozor
* Statusna traka



2‑1 Grafičko sučelje alata Ollydbg

Primjer grafičkog sučelja Ollydbg alata vidljiv je na slici 2-1.

## Meni

Meni pruža pristup glavnim funkcionalnostima alata. On je osnova za korištenje alata. Iako se unutar njega pruža pristup do većine funkcionalnosti alata navođenje svih ponuđenih funkcionalnosti u meni-u značajno bi premašivalo doseg ovog dokumenta. Ipak moguće je navesti glavne funkcionalnosti, odnosno one najčešće korištene, Meni-a Ollydbg alata. Popis ovih funkcionalnosti i pristup istima nalazi se u tablici 2-1.

|  |  |
| --- | --- |
| **Funkcija** | **Pristup preko meni-a** |
| Pokretanje prozora za odabir programa za analizu | File\Open |
| Pokretanje prozora za odabir procesa za analizu | File\Attach |
| Završavanje rada s Ollydbg alatom | File\Exit |
| Otvara log dana prozor | View\Log |
| Otvara executable modules prozor | View\Executable modules |
| Otvara Threads prozor s informacijama o dretvama programa koji se analizira | View\Threads |
| Otvara prozor s prikazom SEH lanca. (Strukturirani upravljač iznimkama) | View\SEH chain |
| Otvara prozor s prikazom postavljenih prekidnih točaka | View\Breakpoints |
| Pokreće program koji se analizira | Debug\Run |
| Pauzira izvođenje programa koji se analizira | Debug\Pause |
| Iz početka pokreće izvođenje programa neovisno o trenutnom stanju analize | Debug\Restart |
| Odspaja Ollydbg od programa koji se trenutno analizira te time završava njegovu analizu | Debug\Close |
| Izvršava trenutnu metodu ulazeći u njeno interno funkcioniranje | Debug\Step into |
| Izvršava trenutnu metodu bez ulaženja u njeno interno funkcioniranje | Debug\Step over |
| Izvršava proces koji se analizira do prvog pojavljivanja return naredbe | Debug\Execute till return |
| Izvršava proces koji se analizira do prvog pojavljivanja korisnički napisanog koda | Debug\Execute till user code |
| Otvara prozor za postavljanje hardware-skih prekidnih točaka | Debug\Hardware breakpoints |
| Otvara prozor za podešavanje izgleda grafičkog sučelja | Options\Appearance |

**Tablica 2-1. Glavne funkcionalnosti Ollydbg alata**

## Glavna alatna traka

U glavnoj alatnoj traci nalaze se tipke koje omogućuju brz i efikasan rad s Ollydbg alatom. One pružaju korisniku brz pristup do najčešće korištenih funkcionalnosti alata. Pregled funkcionalnosti koje se nalaze u glavnoj traci dan je u tablici 2-2.

|  |  |  |
| --- | --- | --- |
| **Ikona funkcionalnosti** | **Ime funkcije** | **Opis funkcije** |
| C:\Users\Petar\Google disk\V.semestar\Projekt_i_PP\Slike\Ollydbg\IkoneAlatneTrake\folderIcon.png | Open next executable | Otvara prozor u kojem možemo izabrati idući program za analizu |
| C:\Users\Petar\Google disk\V.semestar\Projekt_i_PP\Slike\Ollydbg\IkoneAlatneTrake\restarprogramIcon.png | Restart program | Ponovno pokreće program koji se trenutno analizira |
| C:\Users\Petar\Google disk\V.semestar\Projekt_i_PP\Slike\Ollydbg\IkoneAlatneTrake\closeProgramIcon.png | Close program | Prekida analizu programa |
| C:\Users\Petar\Google disk\V.semestar\Projekt_i_PP\Slike\Ollydbg\IkoneAlatneTrake\runProgramIcon.png | Run program | Pokreće program koji se analizira |
| C:\Users\Petar\Google disk\V.semestar\Projekt_i_PP\Slike\Ollydbg\IkoneAlatneTrake\pauseIcon.png | Pause execution | Zaustavlja izvođenje programa koji se analizira |
| C:\Users\Petar\Google disk\V.semestar\Projekt_i_PP\Slike\Ollydbg\IkoneAlatneTrake\stepIntoIcon.png | Step into | Izvršava trenutnu metodu ulazeći u njeno interno funkcioniranje |
| C:\Users\Petar\Google disk\V.semestar\Projekt_i_PP\Slike\Ollydbg\IkoneAlatneTrake\stepOverIcon.png | Step over | Izvršava trenutnu metodu bez ulaženja u njeno interno funkcioniranje |
| C:\Users\Petar\Google disk\V.semestar\Projekt_i_PP\Slike\Ollydbg\IkoneAlatneTrake\traceIntoIcon.png | Trace into | Prati izvođenje programa ulazeći u interno funkcioniranje pozivanih potprograma. |
| C:\Users\Petar\Google disk\V.semestar\Projekt_i_PP\Slike\Ollydbg\IkoneAlatneTrake\traceOverIcon.png | Trace over | Prati izvođenje programa bez ulaženja u interno funkcioniranje pozivnih potprograma |
| C:\Users\Petar\Google disk\V.semestar\Projekt_i_PP\Slike\Ollydbg\IkoneAlatneTrake\executeTillReturnIcon.png | Execute till return | Pokreće izvođenje programa do prve return naredbe |
| C:\Users\Petar\Google disk\V.semestar\Projekt_i_PP\Slike\Ollydbg\IkoneAlatneTrake\goToAdressIcon.png | Go to address in Disassembler | Otvara prozor koji omogućuje pozicioniranje na određenu adresu u kodnom prozoru |
| C:\Users\Petar\Google disk\V.semestar\Projekt_i_PP\Slike\Ollydbg\IkoneAlatneTrake\showLogWindowIcon.png | Show Log window | Otvara Log prozor |
| C:\Users\Petar\Google disk\V.semestar\Projekt_i_PP\Slike\Ollydbg\IkoneAlatneTrake\shoowModuelWindowIcon.png | Show Modules window | Otvara Modules prozor |
| C:\Users\Petar\Google disk\V.semestar\Projekt_i_PP\Slike\Ollydbg\IkoneAlatneTrake\showMemoryWindowIcon.png | Show Memory window | Otvara Memory prozor |
| C:\Users\Petar\Google disk\V.semestar\Projekt_i_PP\Slike\Ollydbg\IkoneAlatneTrake\showThredIcon.png | Show thread | Otvara thread prozor |
| C:\Users\Petar\Google disk\V.semestar\Projekt_i_PP\Slike\Ollydbg\IkoneAlatneTrake\showWindowsIcon.png | Show window | Otvara Windows prozor |
| C:\Users\Petar\Google disk\V.semestar\Projekt_i_PP\Slike\Ollydbg\IkoneAlatneTrake\showHandlesIcon.png | Show handles | Otvara handles prozor |
| C:\Users\Petar\Google disk\V.semestar\Projekt_i_PP\Slike\Ollydbg\IkoneAlatneTrake\showCPUIcon.png | Show CPU | ?? |
| C:\Users\Petar\Google disk\V.semestar\Projekt_i_PP\Slike\Ollydbg\IkoneAlatneTrake\showPatchesIcon.png | Show peaches | Otvara patches prozor |
| C:\Users\Petar\Google disk\V.semestar\Projekt_i_PP\Slike\Ollydbg\IkoneAlatneTrake\showCallStackIcon.png | Show call stack | Otvara call stack prozor |
| C:\Users\Petar\Google disk\V.semestar\Projekt_i_PP\Slike\Ollydbg\IkoneAlatneTrake\showBreakpointsWindowIcon.png | Show breakpoints window | Otvara Breakpoints prozor |
| C:\Users\Petar\Google disk\V.semestar\Projekt_i_PP\Slike\Ollydbg\IkoneAlatneTrake\showReferencesIcon.png | Show references | Otvara references prozor |
| C:\Users\Petar\Google disk\V.semestar\Projekt_i_PP\Slike\Ollydbg\IkoneAlatneTrake\showRunTrace.png | Show run trace | Otvara Run trace prozor |
| C:\Users\Petar\Google disk\V.semestar\Projekt_i_PP\Slike\Ollydbg\IkoneAlatneTrake\showSourceIcon.png | Show source | Otvara Source program |
| C:\Users\Petar\Google disk\V.semestar\Projekt_i_PP\Slike\Ollydbg\IkoneAlatneTrake\debuggingOptionsIcon.png | Debugging options | Otvara Debugging options prozor u kojem je moguće podešavanje opcija pri debugg-iranju |
| C:\Users\Petar\Google disk\V.semestar\Projekt_i_PP\Slike\Ollydbg\IkoneAlatneTrake\apperanceOptionsIcon.png | Appearance options | Otvara prozor za podešavanje izgleda alata |
| C:\Users\Petar\Google disk\V.semestar\Projekt_i_PP\Slike\Ollydbg\IkoneAlatneTrake\helpIcon.png | Help | Otvara prozor koji pruža korisniku pomoć za korištenje programa |

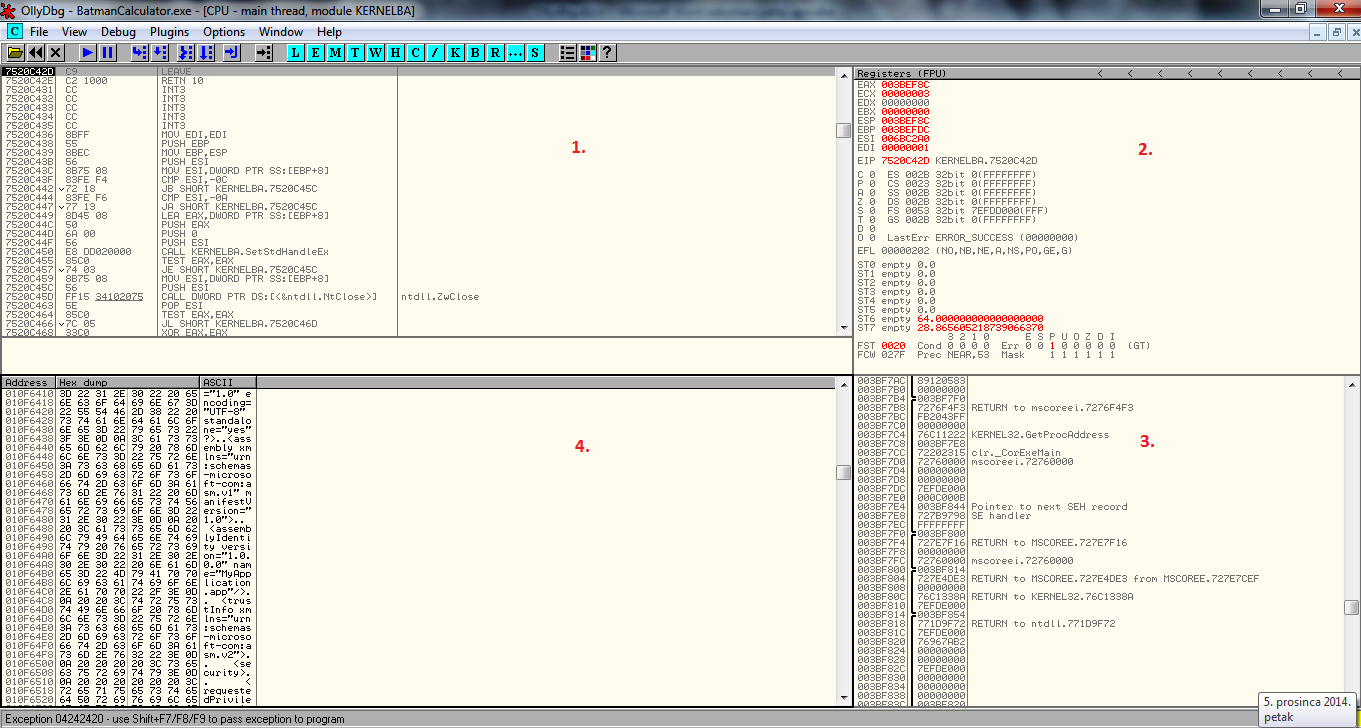
Tablica 2-2. Opcije alatne trake

## Glavni prozor

Glavni prozor Ollydbg-a moguće je podijelit u 4 dijela:

1. CPU / kodni prozor
2. Registarski prozor
3. Prozor stoga
4. Prozor memorijskog spremišta (engl. Memory dump)

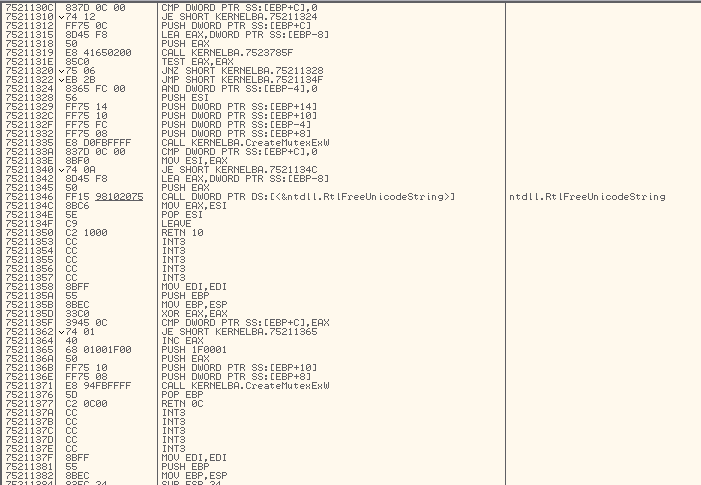
Svaki od ovih prozora prikazuje nam specifične informacije vezane uz program i njegovo izvođenje. Primjer podjele glavnog prozora vidljiv je na slici 2-2.



2‑2 Podjela glavnog prozora Ollydbg-a

### CPU/kodni prozor

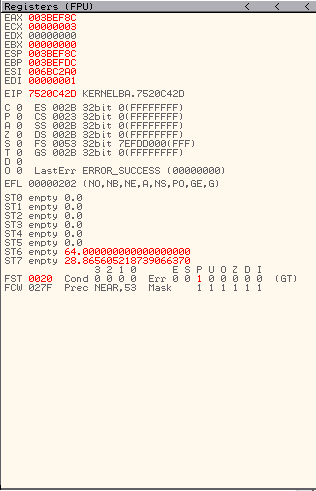
Ovo je najvažniji od 4 dijela glavnog prozora. U ovom prozor nalazi se prikaz asemblerskog koda analiziranog procesa. Za bilo kakav rad s alatom potrebno je barem bazično razumijevanje asemblerskog koda, a za reversni inženjering potrebno je njegovo detaljno poznavanje. Obzirom da je primarna funkcija Ollydbg-a analiza koda ovo je prozor odnosno dio glavnog prozora koji se najviše koristi. Prilikom rada s ovim prozorom postoji više opcija kako je to moguće izvoditi. Moguće je odabiranje naredne te izmjena koja naredba se nalazi na kojem mjestu. Postoji mogućnost unošenja komentara pored prikazanih naredbi. Dodatne opcije za interakciju s podacima prikazanih u kodnom prozor nude se desnim klikom miša na kodni prozor. Primjer CPU/kodnog prozora vidljiv je na slici 2-3.



2‑3 Primjer CPU/ kodnog prozora

### Registarski prozor

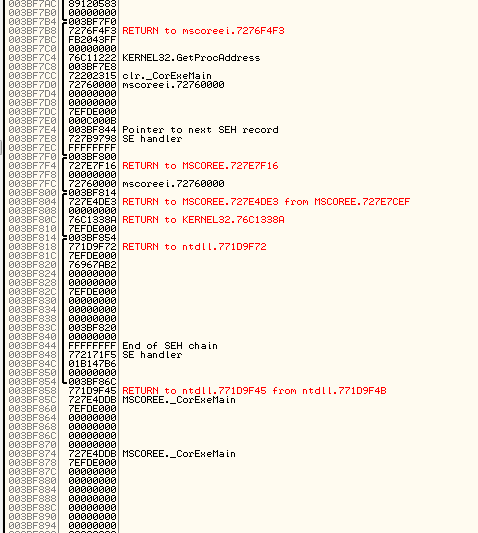
Ovaj registar sadrži informacije o standardnim registrima, zastavicama, dodatne informacije o FPU, MMX, itd. Kako se analizirani proces izvršava sadržaj ovih registara će se očekivano mijenjati. Njihovo praćenje i razumijevanje može biti od izuzetne važnosti prilikom analize procesa. Za dodatnu manipulaciju i prikaz različitih registara dovoljno je kliknuti desni klik i izabrati jednu od opcija ponuđenih u meniju koji se otvori. Primjer registarskog prozora prikazan je na slici 2-4.



2‑4 Primjer registarskog prozora

### Prozor stoga

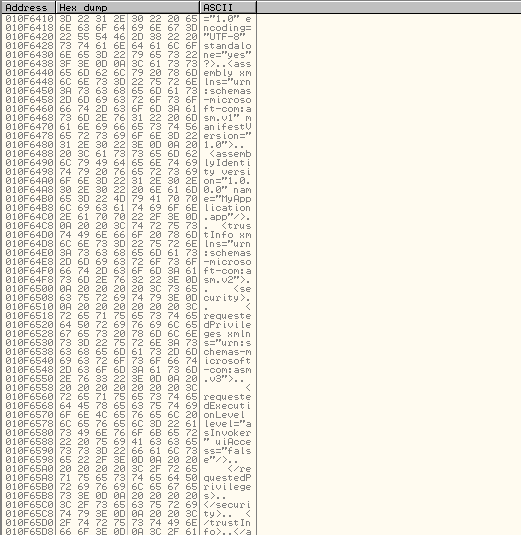
Ovaj prozor sadrži popis adresa dodijeljenih programu prikazanih kao zapis virtualnih adresa ili kao odstupanje od neke specificirane direktne odnosno fizičke adrese. Uz adrese mogu se pojaviti dodatni komentari. Primjer prozora stoga prikazan je na slici 2-5.



2‑5 Primjer prozora stoga

### Prozor memorijskog spremišta

Ovaj prozor prikazuje memorijsko spremište za sve adrese programa. Podaci su prikazani uz ove adrese i mogu biti prikazani primjernom različitih kodnih stranica kao što su ASCII, UNICODE itd. Sve navedene podatke moguće je mijenjati tako što ih se označi te unese željene izmjene. Za dodatne funkcionalnosti manipulacije nad prikazanim podacima na prozoru je dovoljno pritisnuti desni klik mišem kako bi se otvori izbornik dodatnih funkcionalnosti. Jedan redak podataka prikazuje 32 bitovni zapis podataka. Uz kodni prozor ovaj prozor se najčešće koristi prilikom analize. Primjer prozora memorijskog spremišta dan je na slici 2-6.



2‑6 Primjer prozora memorijskog spremišta

## Statusna traka

Statusna traka prikazuje status trenutne analize procesa. S lijeve strane prikazan je savjet za rad u analizi a s desne strane se prikazuje stanje programa koji se analizira. Interakcija sa statusnom trakom nije moguća. Primjer statusne trake prikazan je na slici 2-7.

C:\Users\Petar\Google disk\V.semestar\Projekt_i_PP\Slike\Ollydbg\2-7.png

2‑7 Primjer statusne trake

# Osnove korištenja Ollydbg alata

Početak analize bilo kojeg programa uz pomoć Ollydbg-a je otvaranjem tog programa u Ollydbg-u. Program je moguće otvoriti preko prije spomenute naredbe File\Open ili dovlačenjem .exe datoteke programa na .exe datoteku od Ollydbg-a. Kako bi se olakšao i ubrzao rad praktično j poznavanje globalnih kratica za izvođenje pojedinih naredbi. Popis spomenutih kratica dan je u tablici 3.1.

|  |  |
| --- | --- |
| **Ctrl+F2** | Ponovno pokreće program |
| **Alt+F2** | Zatvara program i završava njegovu analizu |
| **F3** | Nudi izbor otvaranja novog programa za analizu |
| **F5** | Maksimiziran aktivni prozor |
| **F7** | Step into funkcija |
| **F8** | Step over funkcija |
| **F9** | Pokreće program |
| **Shift+F9** | Šalje iznimku obrađivaču iznimci te nastavlja rad programa |
| **Ctrl+F9** | Izvršava kod do prve return naredbe |
| **Alt+F9** | Izvršava kod do korisničkog koda |
| **Ctrl+F11** | Daje podatke o praćenju programa |
| **F12** | Pauzira izvođenje programa |
| **Alt+B** | Otvara prozor s prekidnim točkama |
| **Alt+C** | Otvara CPU prozor |
| **Alt+E** | Otvara Modules prozor |
| **Alt+L** | Otvara Log prozor |
| **Alt+M** | Otvara Memory prozor |
| **Ctrl+T** | Postavljanje uvjeta za zaustavljanje praćenja |
| **Alt+X** | Zatvara OllyDbg |

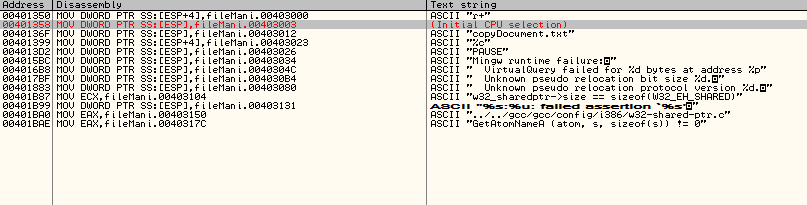
**Tablica 3-1. Globalne kratice**

Za bilo kakav rad odnosno analizu bitno je znati što i kako tražimo. Primjena Ollydbg-a alata je izuzetno široka i nemoguće je u jednom tekstu razmotriti sve njene aspekte. Ipak ono što je moguće je razmotriti najčešće obrasce korištenja i dati smjernice što i kako tražiti prilikom analize.

## Traženje tekstualnih nizova

Traženje tekstualnih nizova javlja se gotovo prilikom svake analize. U korisničkim programima tekstualni nizovi su najčešće oznake na grafičkom sučelju te poruke korisniku, ali isto tako mogu biti i imena datoteka koje program koristi u svom radu, imena domenskih adresa kojima pristupa ili izravno u program upisane šifre.

Učitavanjem i pokretanjem programa u Ollydbg-u on će sam u desnom stupcu kodnog prozora ispisati sve tekstualne nizove koje je identificirao te njihovu kodnu stranicu. Ipak obzirom da analizirani programi mogu često imati previše linija za ljudsko slijedno traženje tekstualnog niza od interesa Ollydbg nudi opciju pretrage za njima. Pritiskom na desni klik povrh kodnog prozora te potom odabirom funkcije Search for\ All referenced text strings otvara se prozor u kojem su slijedno navedeni svi pronađeni tekstualni nizovi. Dodatne funkcije pretraživanja nude se desnim klikom povrh otvorenog prozora. Jednom kad se pronađe tekstualni zapis od interesa pozicioniranje na njega u kodu moguće je izvesti na dva načina. Jedan je očitavanjem adrese na kojoj se on spominje te potom pozicioniranjem na tu adresu preko alatne trake, a drugi preporučljivi je duplim pritiskom na adresu od interesa u prozoru s tekstualnim zapisima. Izgled prozora s tekstualnim zapisima vidljiv je na slici 3-1.



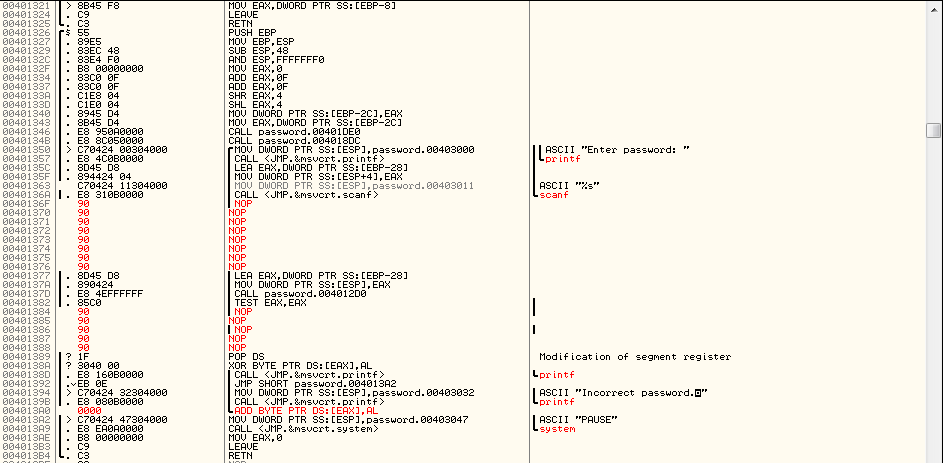
3‑1 Primjer prozora s pronađenim tekstualnim zapisima

## Manipulacija koda

Uz traženje tekstualnih zapisa manipulacija koda je jedna od najčešćih akcija prilikom primjene Ollydbg-a. Ollydbg omogućuje laganu manipulaciju koda. Za izmjenu naredbi u kodu potrebno je izabrati naredbu koja se želi mijenjati te pritisnuti desni klik miša i odabrati opciju Binary/Edit. Uz opciju edit koja omogućuje upisivanje bilo kojeg zapisa nude se mogućnosti unošenja samih nula ili NOP naredbe. Promjene koje se naprave moguće je trajno unijeti u sustav koji se analizira. Najvažnija stvar kod manipulacije koda je smisleno unošenje izmjena, a za to je potrebno poznavanje asemblerskog programiranja. Na slici 3-2 prikazan je primjer koda prije manipulacije, a na slici 3-3 primjer istog koda nakon manipulacije.



3‑2 Primjer koda prije manipulacije



3‑3 Primjer koda nakon manipulacije

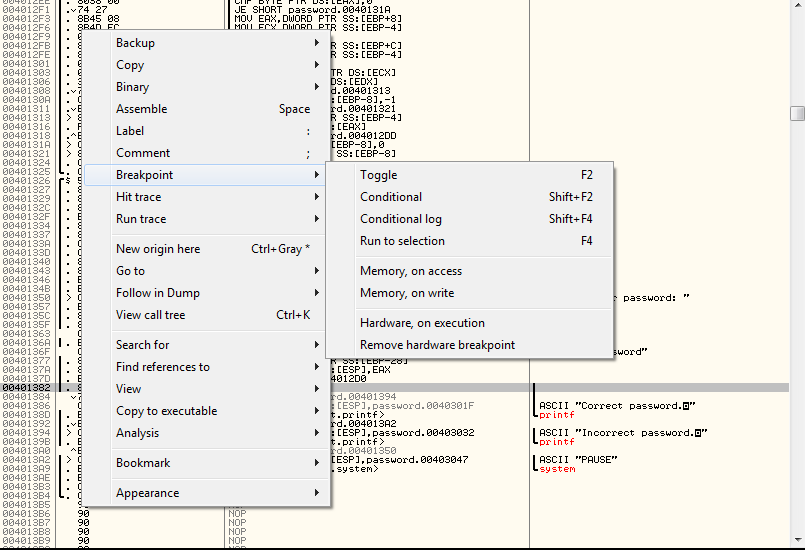
## Postavljanje prekidnih točaka

Postavljanje prekidnih točaka je dio svake analize programa. Svaki, pa čak i sasvim jednostavan, program preveden u asembler zbog kompleksnosti današnjih računala ima izuzetno mnogo naredbi. Kada bi se naredbe išlo analizirati jednu po jednu za analizu bi trebalo previše vremena. Prekidne točke pružaju mogućnost da izvođenje programa bude zaustavljeno pod određenim okolnostima koje sam korisnik definira. Najčešće korištene i najjednostavnije prekidne točke su prekidne točke zadane na pojedinu naredbu. One zaustavljaju izvođenje programa kada se dođe do određene naredbe. U Ollydbg-u je postavljanje takvih prekidnih točaka moguće izvesti duplim klikom miša na adresu naredbe. Jednom kad prilikom izvođenja programa se dođe do te naredbe izvođenje biva zaustavljeno. To omogućuje brzo pozicioniranje na izvođenje koda od posebnog interesa. Primjer postavljanja prekidne točke vidljiv je na slici 3-4.



3‑4 Primjer postavljene prekidne točke

Uz adresne prekidne točke Ollydbg nudi još mnogo naprednih opcija postavljanja prekidnih točaka. Do izbora se dolazi desnim klikom miša iznad kodnog prozora te potom odabirom stavke Breakpoint. Navedeni izbornik vidljiv je na slici 3-5.



3‑5 Izbornik prekidnih točaka

# Ollydbg i analiza zloćudnih programa

Ollydbg je izuzetno često upotrebljavan alat prilikom analize zloćudnih programa. On je upravo savršen za takve primjene jer upravo to su slučajevi kada na raspolaganju nije izvorni kod. Iako Ollydbg ne pruža način za identificiranje koji je program zloćudan on pruža jaku potporu u trenutku kada je određeno koji je program sumnjiv. Učitavanjem sumnjiva programa u Ollydbg moguće je pratiti njegovo izvođenje korak po korak što pomaže identificirati što se točno događa u sustavu odnosno što je to točno što program radi. Iako na prvi pogled može izgledati da Ollydbg pruža toliko informacija o programu i njegovu izvođenu da se, kolokvijalno rečeno, od šume ne vidi stablo to zapravo nije slučaj. Takav privid moguć je onima koji tek počinju raditi s Ollydbg-om no kroz iskustvo slika analize postaje sve bistrija i jasnija. Za provođenje analize potrebno je dobro poznavanje sustava na kojem se program izvršava. Kako je raspon djelovanja zloćudnih programa izuzetno širok teško je u jednom tekstu navesti eksplicitne detalje analize ipak moguće je ponuditi generalne upute.

Prilikom analize sumnjiva programa dobro je za početak odrediti:

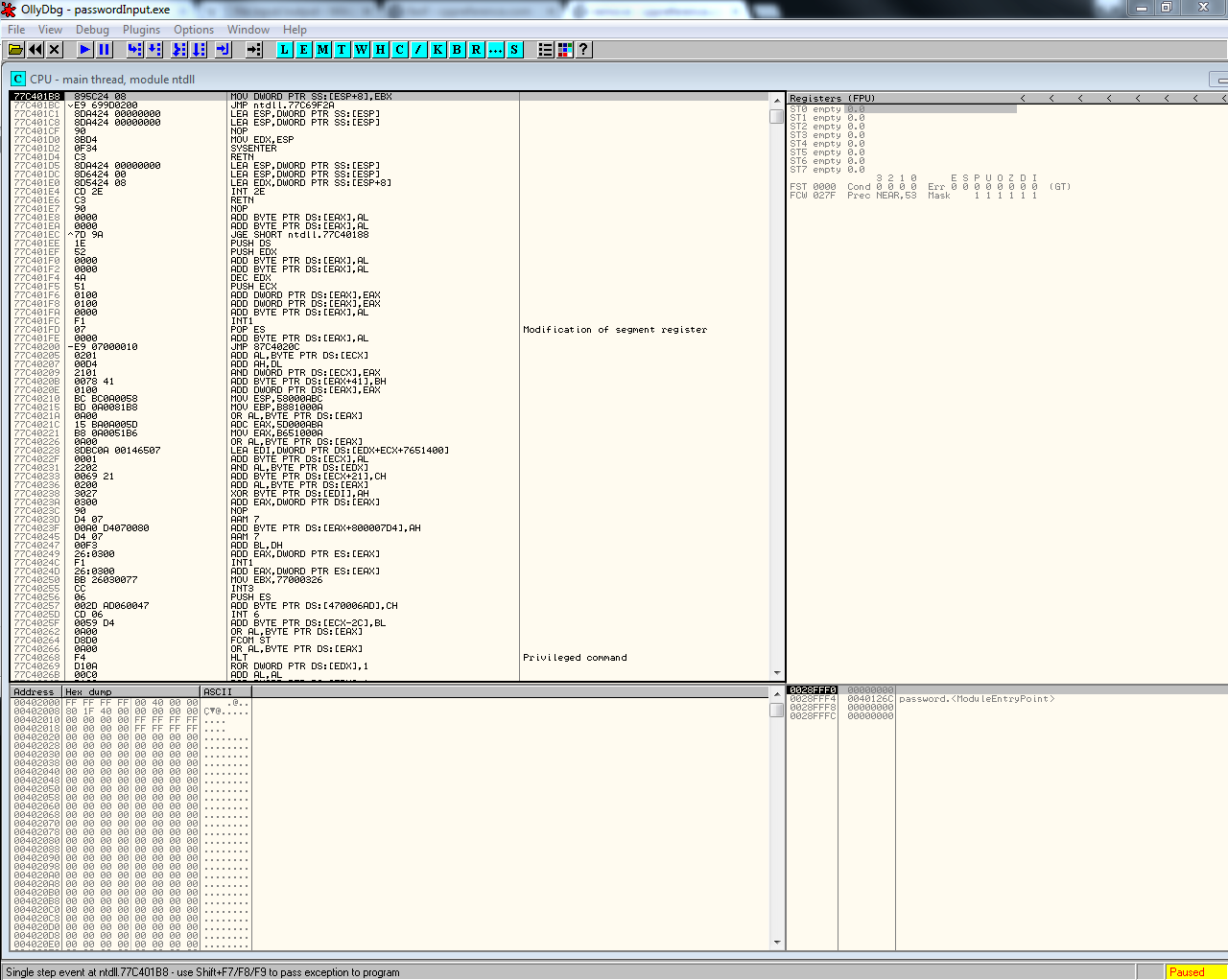
* Da li program komunicira s mrežom? Ukoliko sumnjivi program komunicira s mrežom potrebno je računalo na kojem se provodi analiza od spojiti od mreže te tek nakon toga nastaviti analizu. Nakon što se od spoji računalo s mreže, potrebno je identificirati kojim adresama na mreži pristupa. Postoji dobra mogućnost da su takve informacije negdje u programu zapisane u obliku tekstualnog niza. Ako se identificiraju mrežne adrese s kojima program komunicira i primijeti se da je za pristup istima potrebna autentifikacije može se pokušati identificirati autentifikacijske informacije. Idući korak je identificirati koje su informacije slane na mrežu.
* Da li program čita i/ili piše po nekim datotekama ili sistemskim postavkama? Ukoliko se ustanovi da program čita i/ili piše po nekim datotekama potrebno je utvrditi o kojim se datotekama radi odnosno kakve informacije one sadrže. Informacije o tome po kojim datotekama program čita moguće je naći u programu tijekom izvođenja ili čak upisano kao tekstualan niz. Ako se ustanovi da program piše odnosno prepravlja sistemske podatke potrebno je ustanoviti napravljenu štetu nad njima.

Dvije navedene točke preporučljivo su dio svake analize. Nakon što se njih provede može se krenuti u detaljniju analizu programa, a kasnije i u prepravljanje programa kako bi poslužio u korist onog tko ga analizira.

# Primjeri analize - Pristup ograničenom dijelu programa

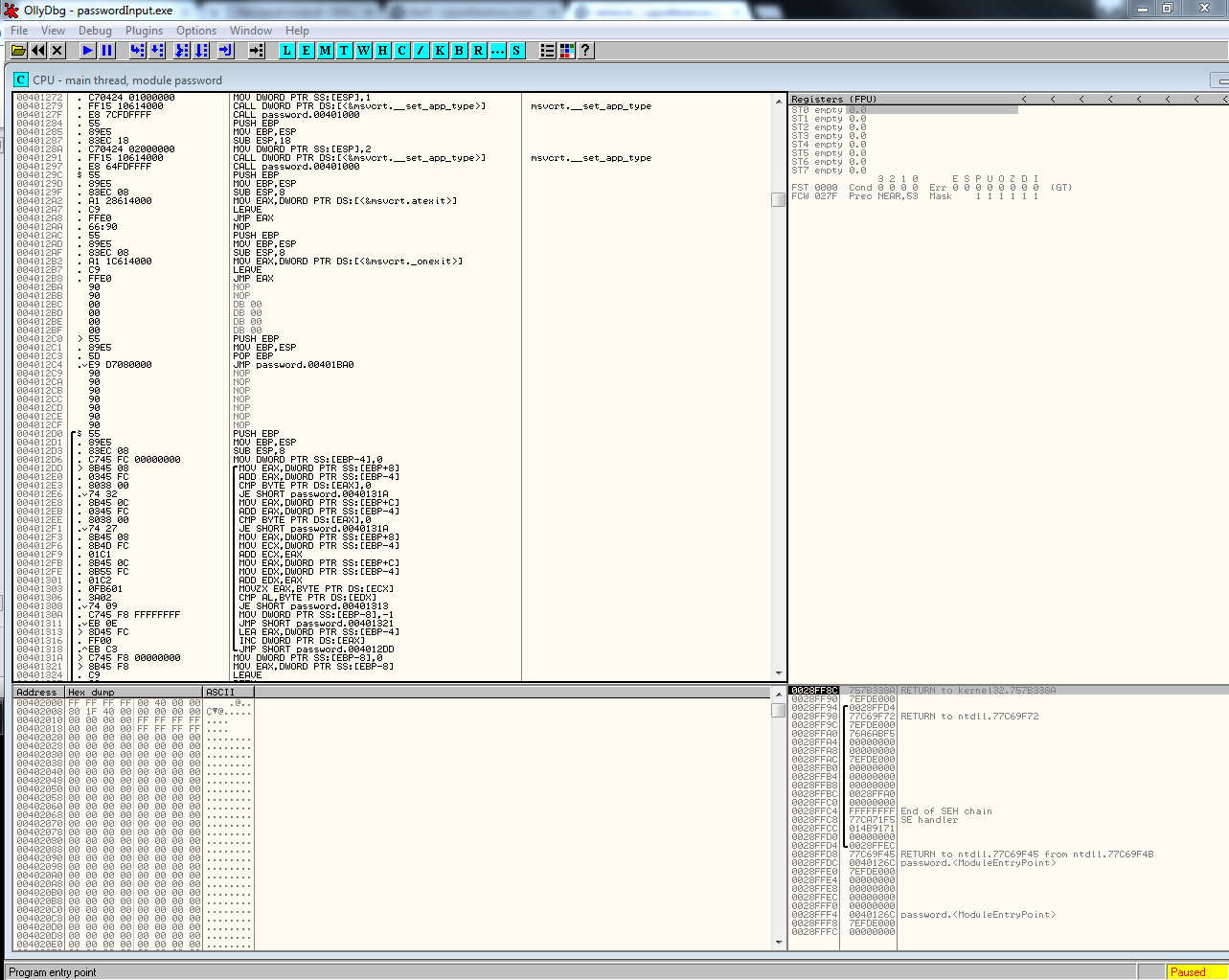
Za demonstraciju jednostavne analize programa razmatra se jednostavan program koji od korisnika traži učitavanje šifre jednom kad program dobije ispravnu šifru on prestaje s radom. Iako navedeni program nema nikakvu malicioznu funkciju on prikazuje jednu od često viđenih funkcionalnosti zloćudnih programa, a to je restrikcija određenom dijelu programa i određenim podacima u njemu. Isto kao što se u ovoj analizi pokazuje pristup do pronalaska šifre odnosno do statusa odobrenja tako se prilikom bilo koje druge analize može primjenjivati iste korake analize za identifikaciju ključnih informacija u programu.

Analiza započinje učitavanjem programa u Ollydbg. Nakon učitavanja stanje u alatu prikazano je na slici 5-1.



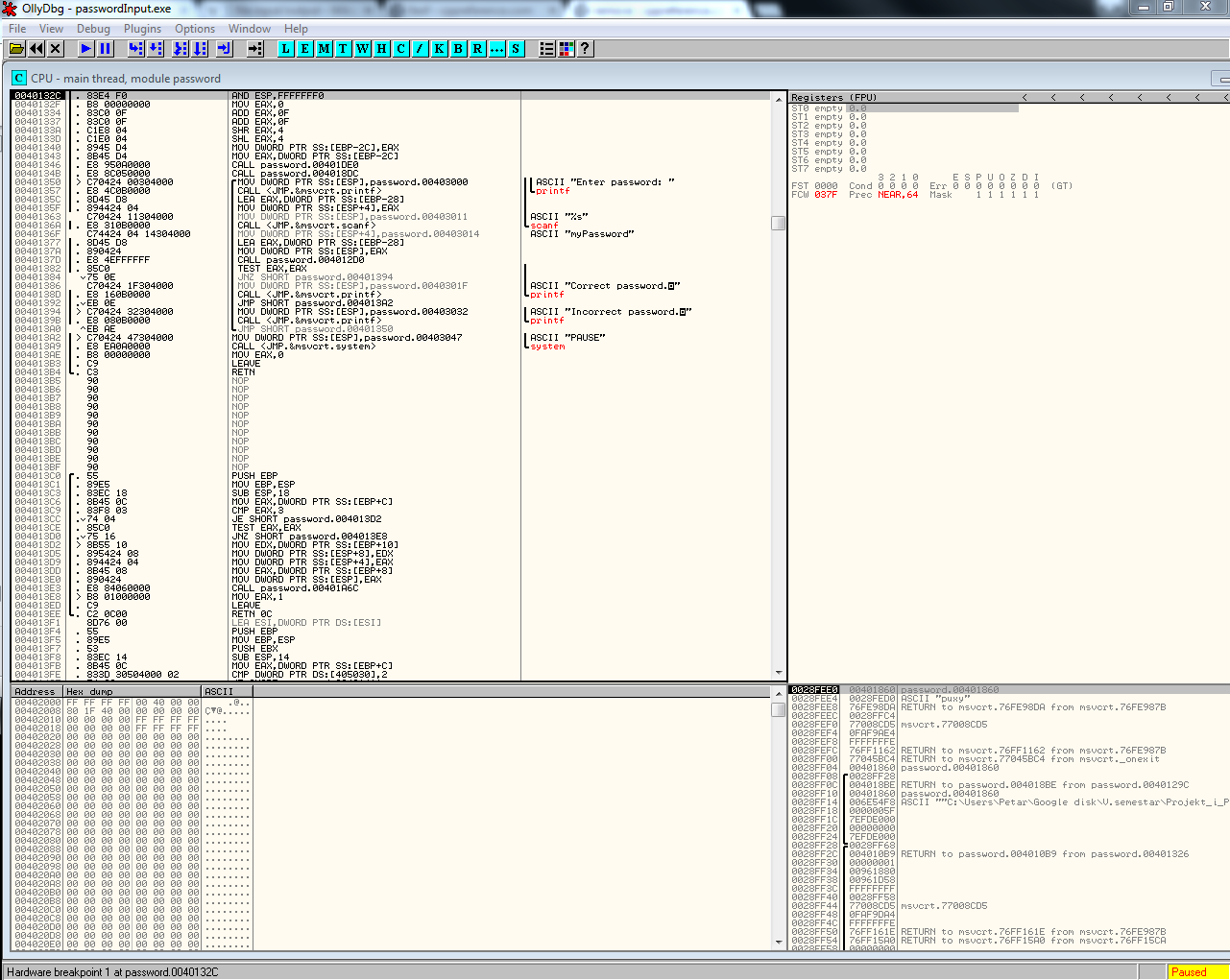
5‑1 Stanje nakon učitavanja programa koji se analizira

Nakon što je program učitan potrebno ga je pokrenuti. Prvim pritiskom na tipku Run program će se dovesti do trenutka izvođenja korisničkog programskog koda. Stanje analize nakon ovog koraka prikazano je na slici 5-2.



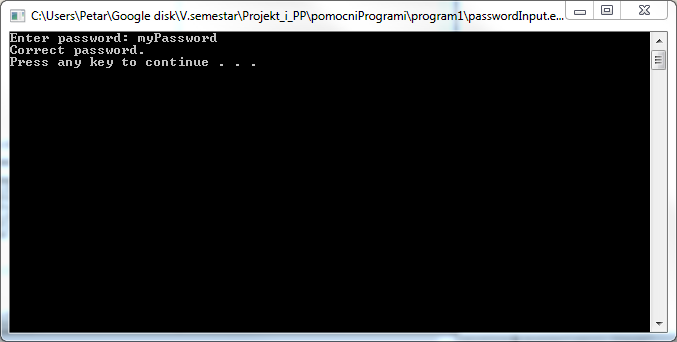
5‑2 Stanje nakon pokretanja programa koji se analizira

Nakon što je program pokrenut još jednim pritiskom na tipku run počinje s izvršavanjem korisničkog koda te dolazi do dijela na kojem od korisnika traži unos šifre. Stanje analize vidljivo je na slici 5-3.



5‑3 Stanje analiziranog programa nakon izvršavanja dijela korisničkog koda

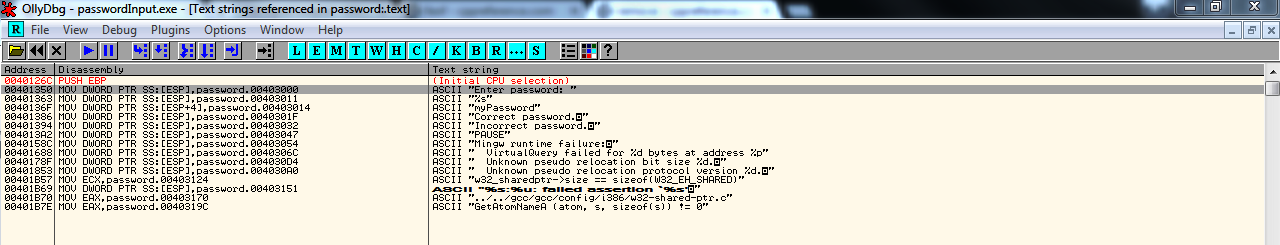
Na slici 5-3 je vidljivo da je Ollydbg prepoznao zapise ispisnog teksta „Enter password“ te poruke koje se ispisuju ovisno o točnosti unesenih podataka. Isto tako moguće je primijetiti i spominjanje tekstualnog niza „myPassword“ koji se nigdje u interakciji s korisnikom ne spominje. Obzirom da se spomenuti niz ne spominje pri interakciji s korisnikom logično ga je isprobati kao traženu šifru. Rezultat unosa vidljiv je na slici 5-4.



5‑4 Rezultat unos tekstualnog niza "myPassword"

Vidljivo je unos pronađenog niza uspio. Sad se nalaže pitanje što da je program koji se analizira bio veći te da je koristio nekakvo kriptiranje tako da Ollydbg nije u stanju direktno prepoznati navedeni tekstualni niz. Na još jednoj analizi istog programa prikazan je postupak u tom slučaju.

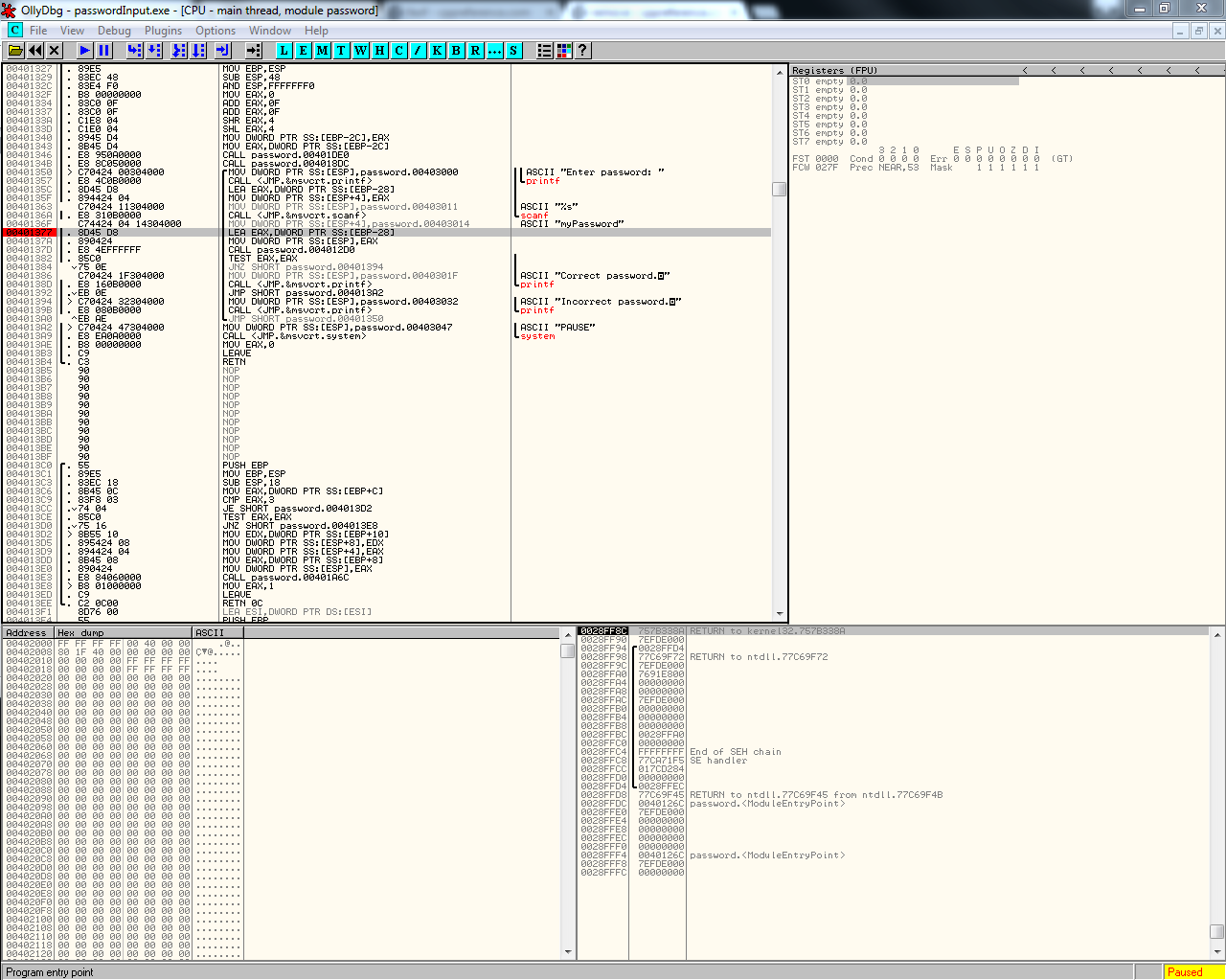
Program se ponovno pokreće do izvođenja programskog koda. Iako je ovaj program linearan i njegovo izvođenje uvijek vodi do unosa šifre za slučaj da to nije tako do navedenog dijela je moguće doći traženjem tekstualnog niza koji se ispisuje korisniku pri traženju unosa. Primjer stanja pretrage teksta u programu prikazan je na slici 5-5.



5‑5 Rezultat pretrage tekstualnih zapisa u programu

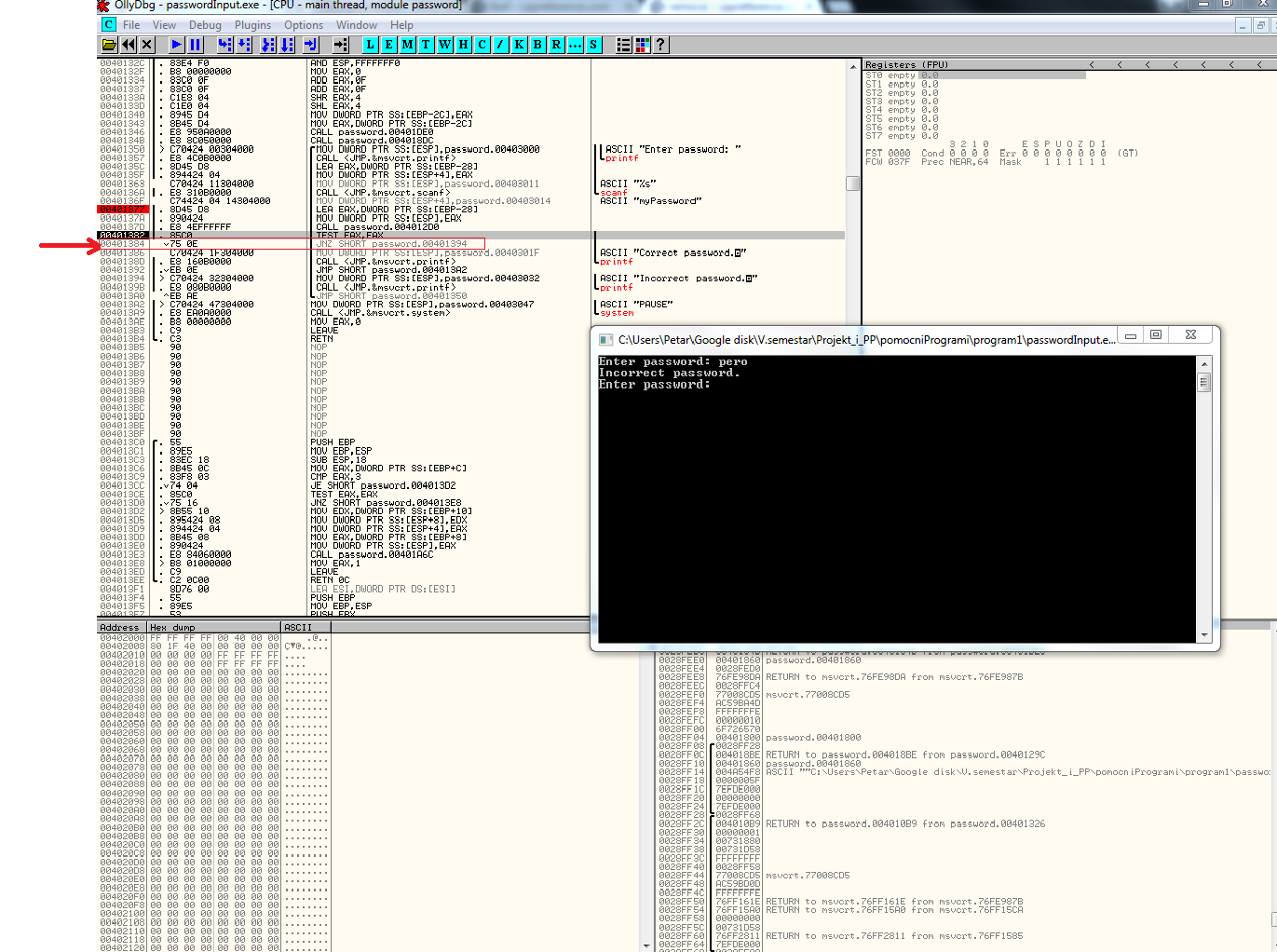
Nakon što je pronađen tekstualni zapis od interesa na njega se u kodu moguće pozicionirati pritiskom na adresu navedenu uz njega. Iako je izgled sučelja sada isti kao na slici 5-3 ostatak analize provodi se kao da ne postoji ispis niza „myPassword.“

Analizom ispisanog asemblerskog koda moguće je uvidjeti da postoje skokovi koji se temelje na nekim usporedbama te usmjeravaju hoće li program prihvatiti uneseni niz ili ne. Isto tako Ollydbg sa strane ispisuje prepoznate naredbe ispisa i upisa teksta. Za danju analizu potrebno je odrediti koji skok vodi ka prihvaćanju ulaznog niza odnosno njegovom ne prihvaćanju. To je najbolje napraviti tako da se nakon naredbe scanf postavi prekidna točka te se potom izvođenje programa promatra korak po korak. Primjer postavljene prekidne točke vidljiv je na slici 5-6.



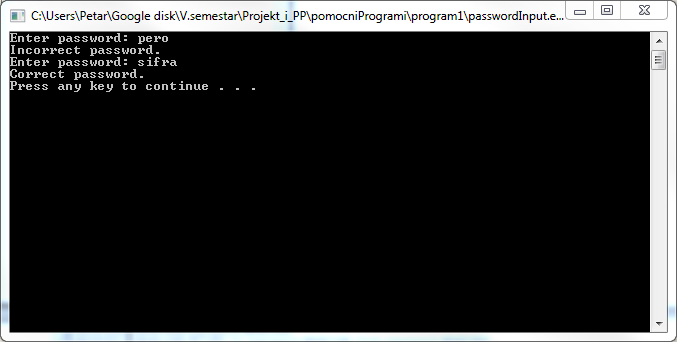
5‑6 Primjer postavljanja prekidne točke

Nakon postavljanja prekidne točke dopušta se danji rad programa. Unosi se nasumična šifra te se rad programa promatra korak po korak. Ovom analizom utvrđena je naredba skoka koja uzrokuje preskakanje dijela koda koji prihvaća uneseni niz. Prikaz identificirane naredbe skoka vidljiv je na slici 5-7.



5‑7 Identificirana naredba skoka

Nakon identificiranja krucijalne naredbe skoka vidljivo je da ukoliko ona ne bi postojala program bi prihvatio svaki unos. Iz tog razloga navedenu naredbu skoka mijenja se s NOP. Potom se ponovno unos bilo kakav tekstualni niz kao šifra. Rezultat izvođenja programa vidljiv je na slici 5-8.



5‑8 Rezultat izvođenja programa nakon izmjena koda

Očito je da u ovom slučaju poznavanje šifre nije bilo potrebno već je program nasilno usmjeren na prihvaćanje unesenog niza.

# Zaključak

Ollydbg je zasigurno najmoćniji alat za analizu programskog koda, barem od onih slobodno dostupnih. Mogućnosti manipulacije i analize izvršnih datoteka koje on pruža su i više nego dorasle ozbiljnim zahtjevima sofisticirana analize. Intuitivno grafičko sučelje i dobra organizacija podataka uvelike korisniku olakšavaju njegovo korištenje. Težina Ollydbg-a ne leži u njemu kao takvom već u području za koje je namijenjen. Za korištenje je potrebno poznavanje asemblera, operacijskih sustava, prevođenje programskih jezika i još mnogo detalja. Čitatelju kojem navedeni tekst predstavlja prevelik problem pri razumijevanju preporuča se bolje izučavanje područja računarstva prije samog rada s Ollydbg-om.

Iako se program često koristi pri analizi zloćudnih programa daleko je češće korišten prilikom uklanjanja programskih zaštita na komercijalnim programima. Zbog njegove slobodne dostupnosti izuzetno je popularan među računalnim kriminalcima te zbog toga postoje inicijative da se njegova dostupnost ograniči.

# Literatura

[1] Ollydbg : <http://www.ollydbg.de/>

[2] Fakultet elektrotehnike i računarstva; Radovi iz područja računalne sigurnosti; 2014; <http://os2.zemris.fer.hr/index.php?show=start>