CARNET

Ghidra

CERT.hr-PUBDOC-2021-1-401

CERT.hr

Sadržaj

1 Uvod

Svakodnevno korištenje Interneta, preuzimanje datoteka i programa, korištenje USB-ova na različitim nesigurnim računalima i slične aktivnosti dovode korisnika u kontakt s raznim zlonamjernim programima.

Razni antivirusni softveri i besplatni *online* alati ne zahtijevaju veliko stručno znanje za korištenje, a u većini slučajeva će svojom automatiziranom statičkom i dinamičkom analizom uspjeti dati korisniku točan odgovor na pitanje je li neki program zlonamjeran.

No, često se javlja potreba za još dublji i detaljniji uvid u ponašanje nekog programa, tj. želimo ustanoviti što točno taj program radi na računalu kako bismo došli do određenih zaključaka. Takav proces detaljne analize na niskoj razini naziva se reverzno inženjerstvo i u njemu pomažu alati poput OllyDbg, IDA-e, x64dbg itd., od kojih su neki komercijalni, a neki besplatni i/ili otvorenog koda.

Ghidra je jedan od najnovijih alata otvorenog koda namijenjenih reverznom inženjerstvu. Razvila ga je američka NSA (*engl. National Security Agency*), a javno je objavljen 5. ožujka 2019 godine.

U ovom dokumentu objasnit će se instalacija i korištenje, Ghidre, pokazati primjer analize zlonamjernog programa i navesti neke korisne funkcionalnosti za analizu programa.

Za razumijevanje ovog dokumenta pretpostavlja se da je čitatelj upoznat s pojmom i procesom reverznog inženjerstva softvera. Drugim riječima, onoga tko ne zna 'reversati' ovaj dokument to neće ni naučiti. Ovaj dokument oslanja se na primjenu i funkcionalnosti Ghidre kao alata u reverznom inženjerstvu, a ne reverzno inženjerstvo općenito.

2 Instalacija alata Ghidra

Alat Ghidra razvija američka Nacionalna sigurnosna agencija (engl. *National Security Agency*, NSA) i može se preuzeti sa službene stranice alata.

Ghidra podržava brojne operacijske sustave:

- Microsoft Windows 7 ili 10 (64-bit)
- Linux (64-bit, CentOS 7 je poželjan)
- macOS (OS X) 10.8.3+ (Mountain Lion ili novije)

te zahtijeva sljedeće sistemske značajke:

- Hardver:
    - 4GB RAM
    - 1 GB memorije za pohranu
    - preporučena su dva monitora
- Softver:
    - Java 11 Runtime and Development Kit (JDK)
        - preporučen je OpenJDK s jdk.java.net

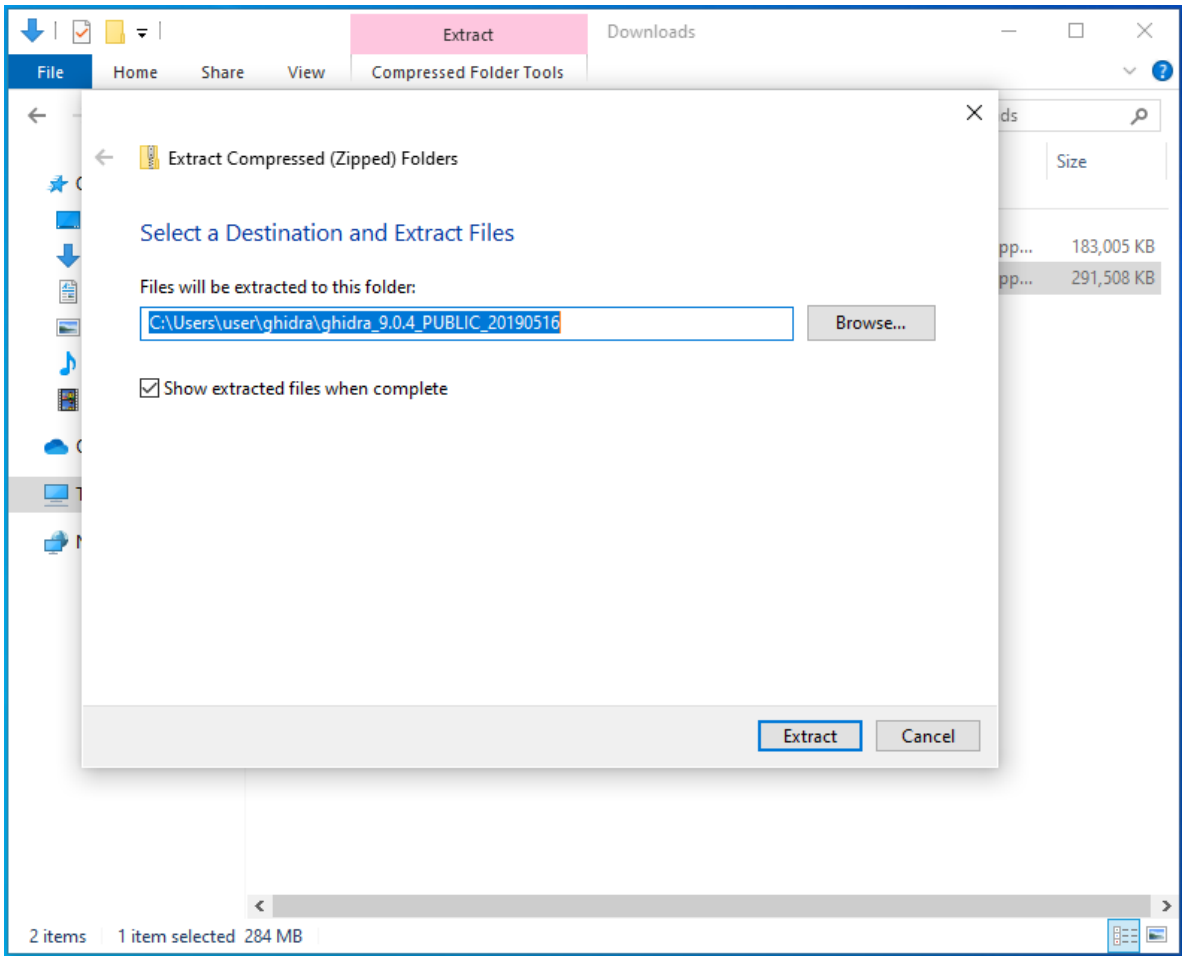
Ove instrukcije napisane su za Ghidru 9.0.4 i operacijski sustav Windows 10. Slične instrukcije, na engleskom jeziku i za ostale operacijske sustave, mogu se pronaći na službenim instrukcijama za instalaciju.

Prvi korak je preuzeti Ghidru.

Slika 1 *Ghidrina službena web stranica*

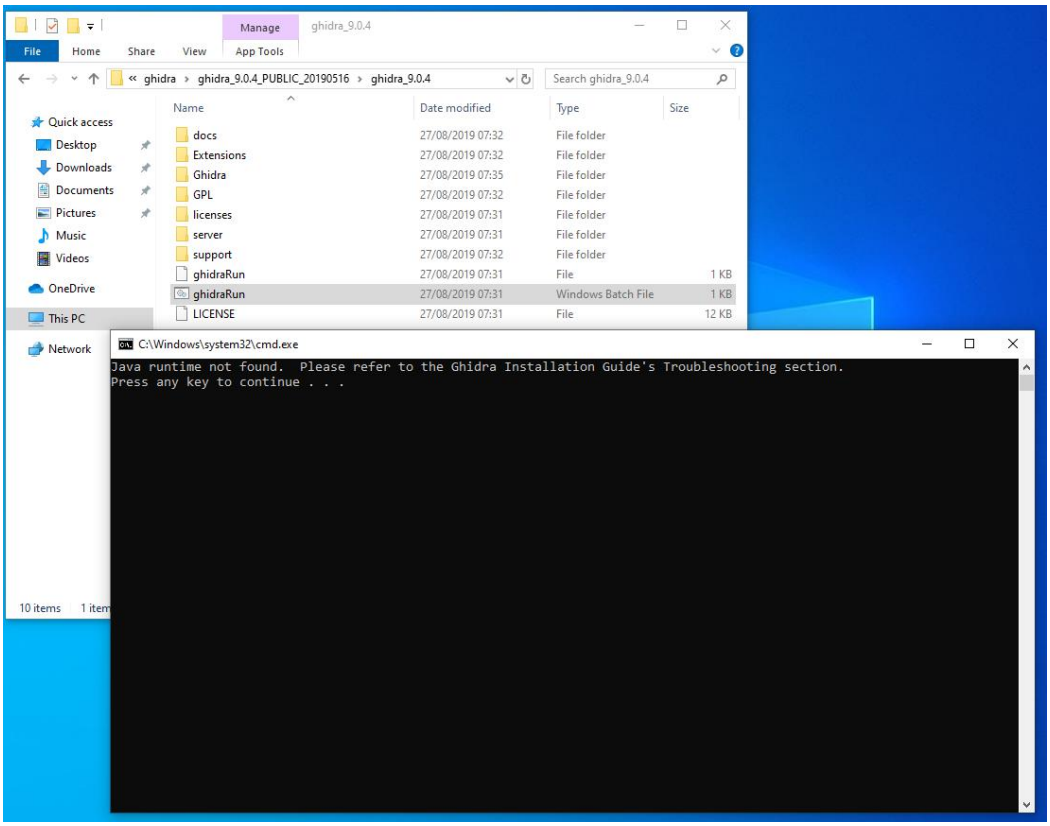
Ovdje je korisno napomenuti da Ghidra ne koristi tradicionalni instalacijski program, nego se samo raspakira u datotečni sustav i pokrene. Ovaj način omogućuje instalaciju bez administratorskih privilegija, ali nažalost ne može automatski stvoriti prečac (engl. *shortcut*) na radnoj površini (engl. *desktop*) ili u početnom izborniku (engl. *start menu*).

Nakon preuzimanja, potrebno je raspakirati preuzetu zip arhivu te odabrati lokaciju gdje će se Ghidra pohraniti. Preporučeno je odabrati mapu nalik „C:\Users\user\ghidra" ili, u slučaju da imate administratorske privilegije, u „C:\Program Files". Naravno, može se proizvoljno odabrati i neka druga mapa.



Slika 2 *Otpakiravanje Ghidre*

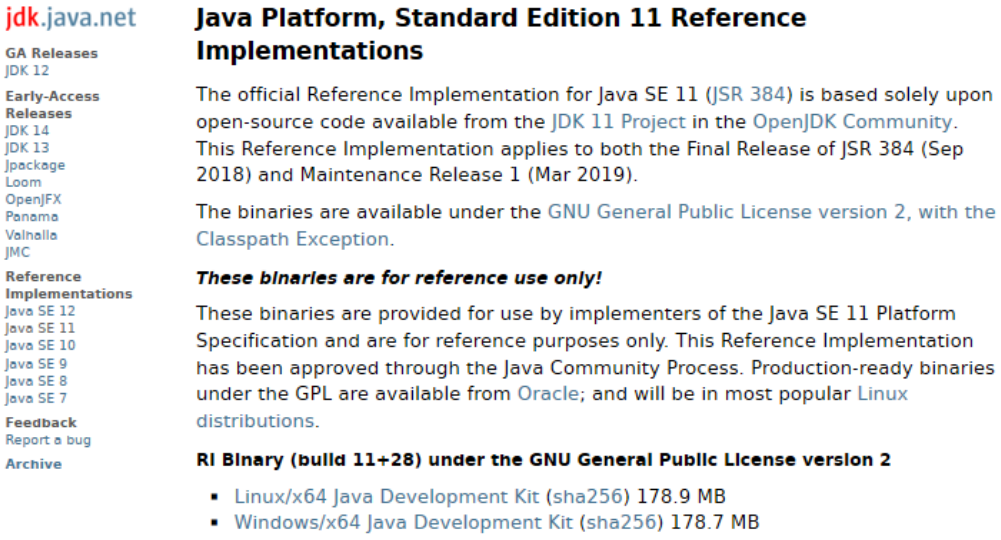
Nakon što ste uspješno raspakirali Ghidru, pozicionirajte se u izabrani direktorij te je pokušajte pokrenuti dvostrukim klikom na skriptu „*ghidraRun.bat*".



Slika 3 *Pokušaj pokretanja Ghidre*

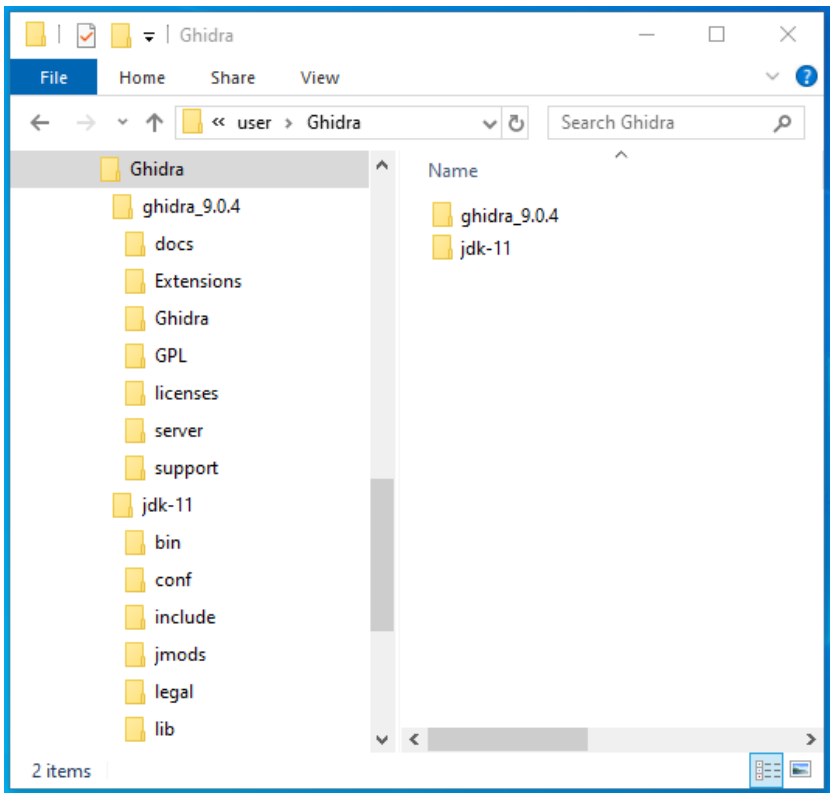
U slučaju da se Ghidra uspješno pokrenula, vrlo vjerojatno imate već otprije instaliran Java 11 JDK te možete preći na instrukcije za korištenje alata. U suprotnom, pratite daljnje upute.

Preuzmite OpenJDK 11 sa službene stranice jdk.java.net.



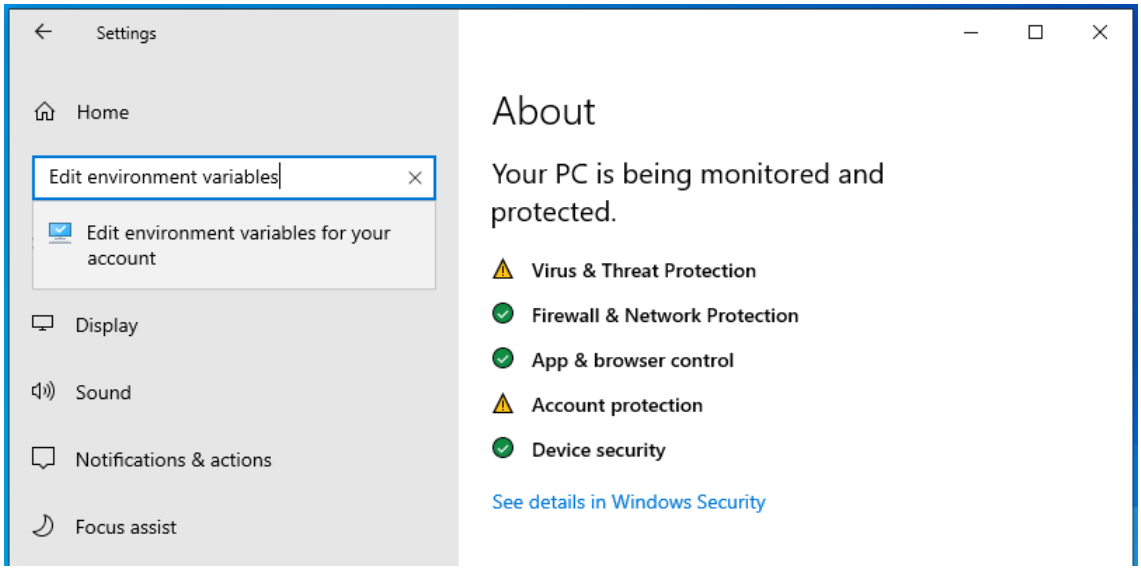
Slika 4 *Službena stranica OpenJDK-a*

Nakon preuzimanja, potrebno je raspakirati preuzetu zip arhivu. Preporučeno je odabrati isti direktorij gdje ste raspakirali i Ghidru (u našem slučaju: „C:\Users\user\Ghidra").



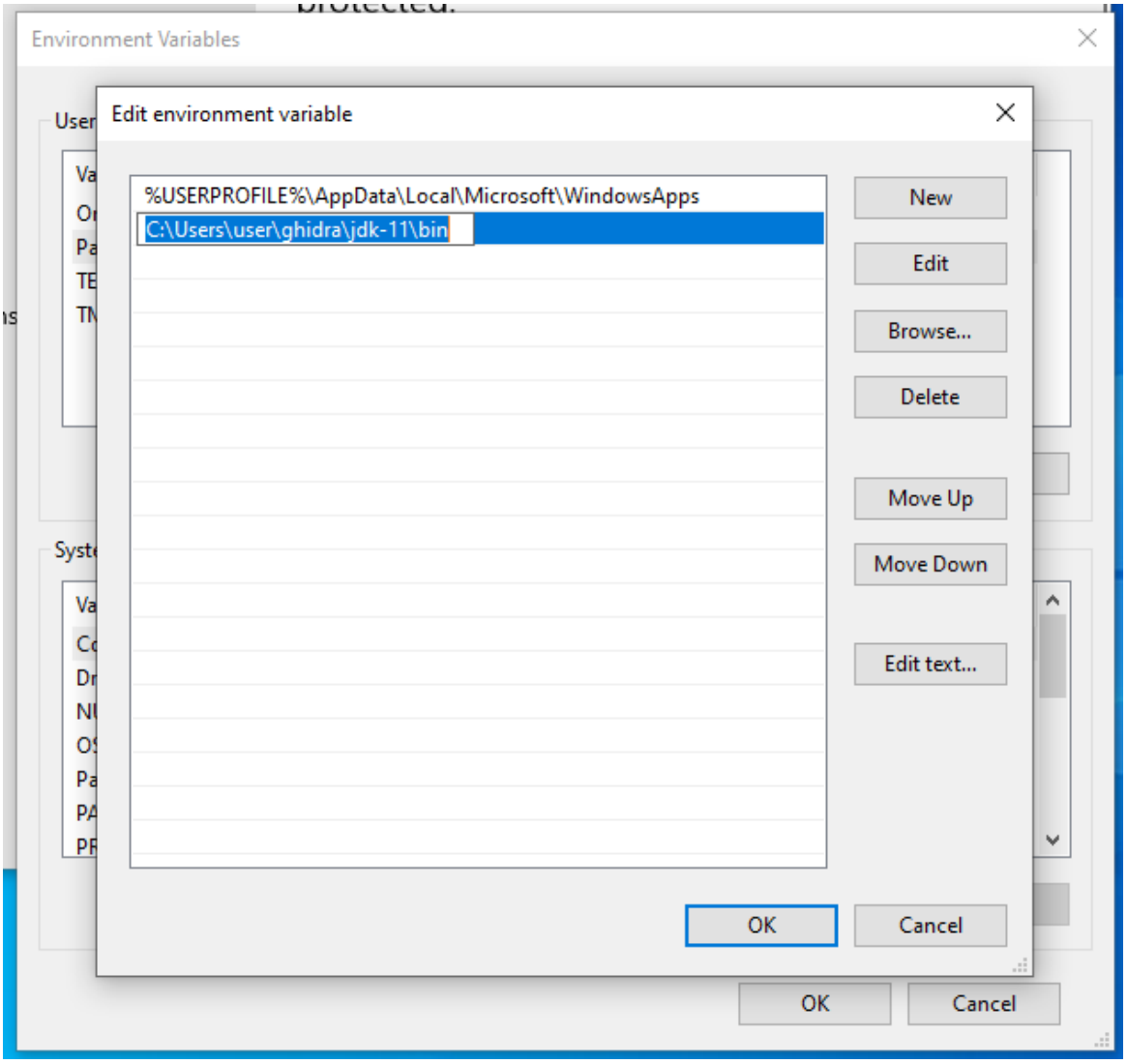
Slika 5 *Direktorij „C:\Users\user\Ghidra" nakon raspakiravanja OpenJDK 11*

Da bi Ghidra mogla koristiti OpenJDK 11, potrebno je dodati varijablu okoline PATH (engl. *enviroment path variable*). Nakon desnog klika na početni izbornik, odaberite stavku sustav (engl. *system*). Potražite „*Edit environment variables for your account*" te kliknite na prvi ponuđeni rezultat.



Slika 6 *Uređivanje korisničkih varijabli*

Unutar novootvorenog prozora „*Enviroment variables*" odaberite korisničku varijablu PATH te kliknite „*Edit*". Nakon toga, u novom prozoru kliknite tipku „*New*" te upišite „<lokacija otpakiranog OpenJDKa 11>\bin", u našem slučaju „C:\Users\user\Ghidra\jdk-11\bin". Potvrdite dodavanje nove PATH varijable klikom na tipku „*Ok*".



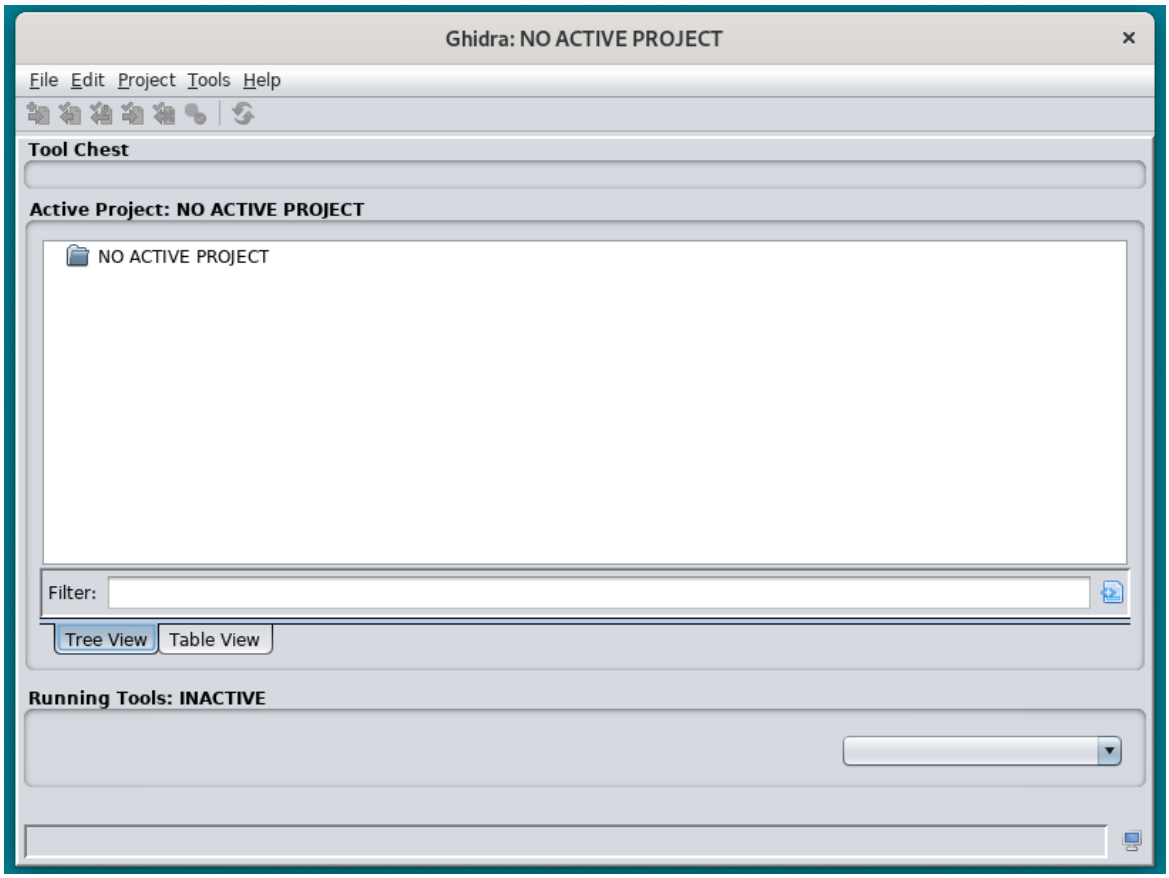
Slika 7 *Dodavanje korisničke varijable*

Sad, ako ponovno navigirate u direktorij gdje ste otpakirali Ghidru te ju ponovno pokušate pokrenuti dvostrukim klikom na „*ghidraRun.bat*", trebali biste uspješno pokrenuti program. Otvorit će vam se prozor s uvjetima korištenja te, nakon što ih pročitate i prihvatite, možete koristiti program.

3 Korištenje alata Ghidra

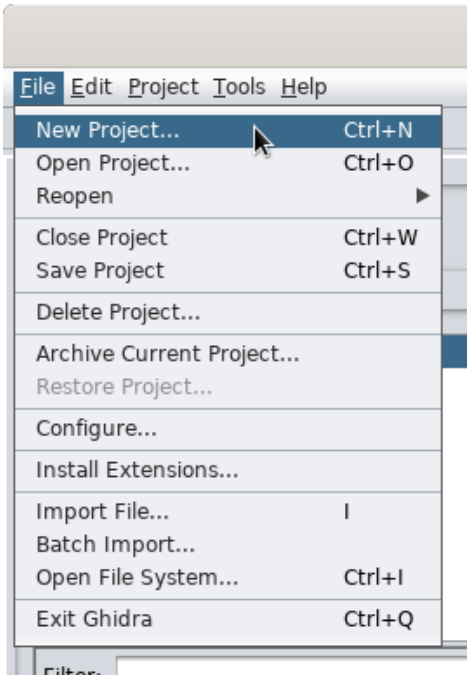
3.1 Osnove Ghidre

Ghidra se pokreće dvostrukim klikom na „*ghidraRun.bat*”. Ako će se Ghidra redovito koristiti, preporučeno je stvoriti prečac na radnoj površini za lakše pokretanje. Nakon što pokrenete Ghidru pokazat će vam se savjet dana, kojeg slobodno zatvorite da vidite prozor za odabir projekta.

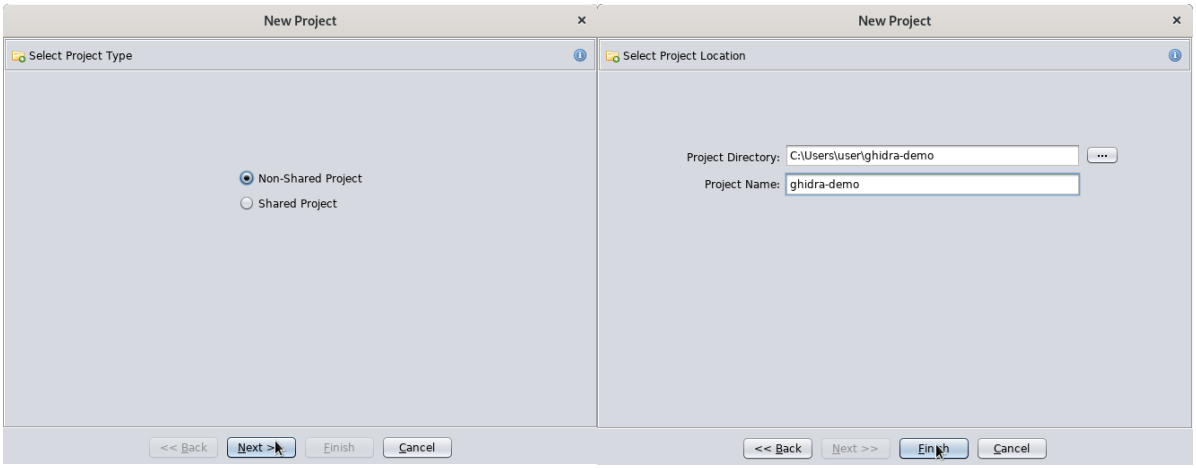


Slika 8 *Ghidrin početni ekran, prozor za odabir projekta*

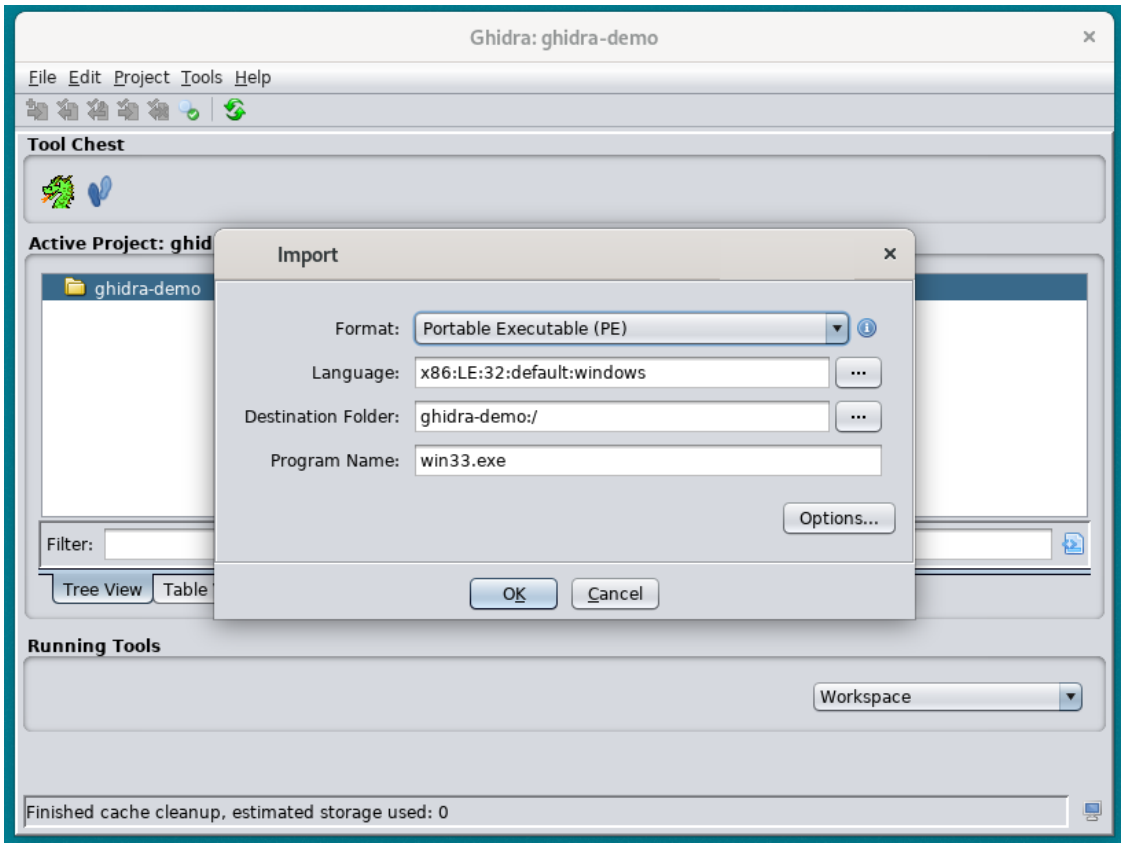
Trenutno ne postoji nijedan aktivan projekat za odabrati pa morate stvoriti novi. Novi projekt možete stvoriti klikom na „*File->New Project...*” ili koristeći prečac na tipkovnici „Ctrl+N”, što će pokrenuti stvaranje novog projekta. Najprije ćete morati odabrati želite li započeti dijeljeni projekt za rad u timu (engl. *Shared Project*) ili ćete raditi sami na vlastitom projektu (engl. *Non-Shared project*). Odaberite vlastiti projekt. Kliknite „*Next* >>” te odaberite proizvoljnu lokaciju i naziv vašeg projekta (npr. lokaciju: „C:\Users\user\ghidra-demo”, naziv: „ghidra-demo”) te kliknite „*Finish*”.



Slika 9 *Stvaranje novog projekta*



Sada ste spremni uvesti (engl. *import*) program koji će se analizirati. Program je moguće uvesti odabirom „*File->Import File...*”, tipkovnim prečacem „I” ili jednostavnim povlačenjem i ispuštanjem datoteke na odgovarajući projekt. Nakon odabira vašeg programa, Ghidra će sama pokušati prepoznati format i mikroprocesorski jezik. Ako ste zadovoljni s odabirom kliknite „*OK*” i pokrenut će se proces uvoza.

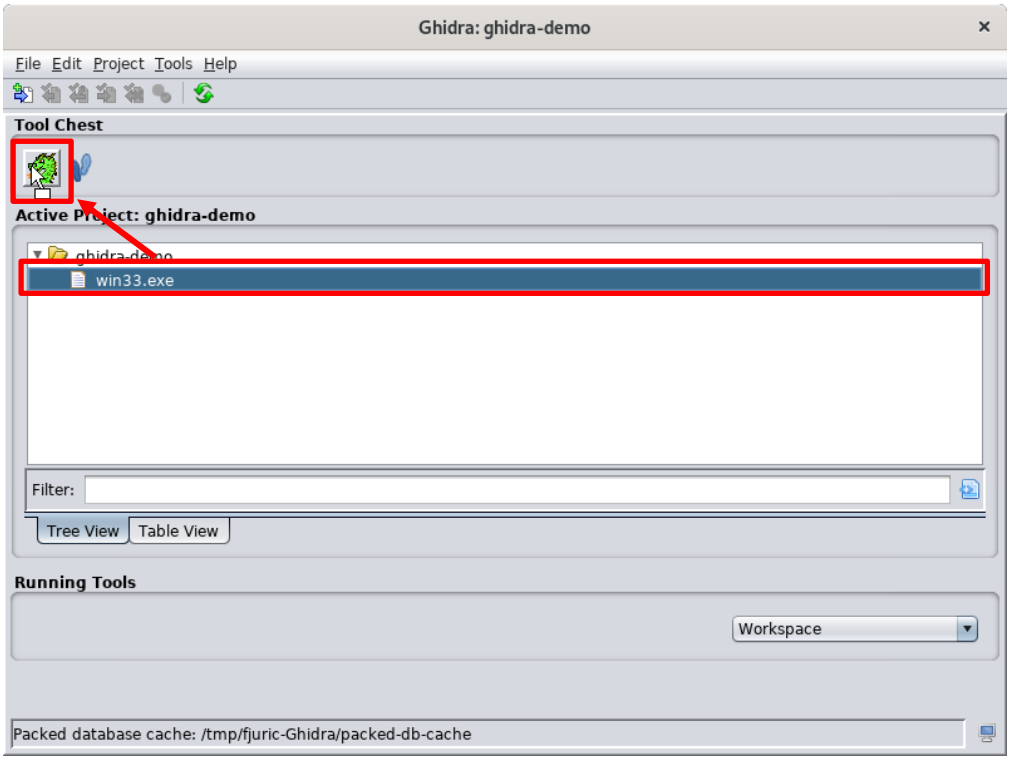


Slika 10 ***Proces uvoza datoteke***

Uvest ćemo zlonamjerni program (engl. *malware*) naziva „Dexter" koji je u 2012. godini napadao blagajne u više od 40 zemalja te krao osjetljive informacije poput brojeva kartica. Važno je napomenuti da zlćudne programe treba raspakirati i analizirati samo unutar izoliranog virtualnog stroja.

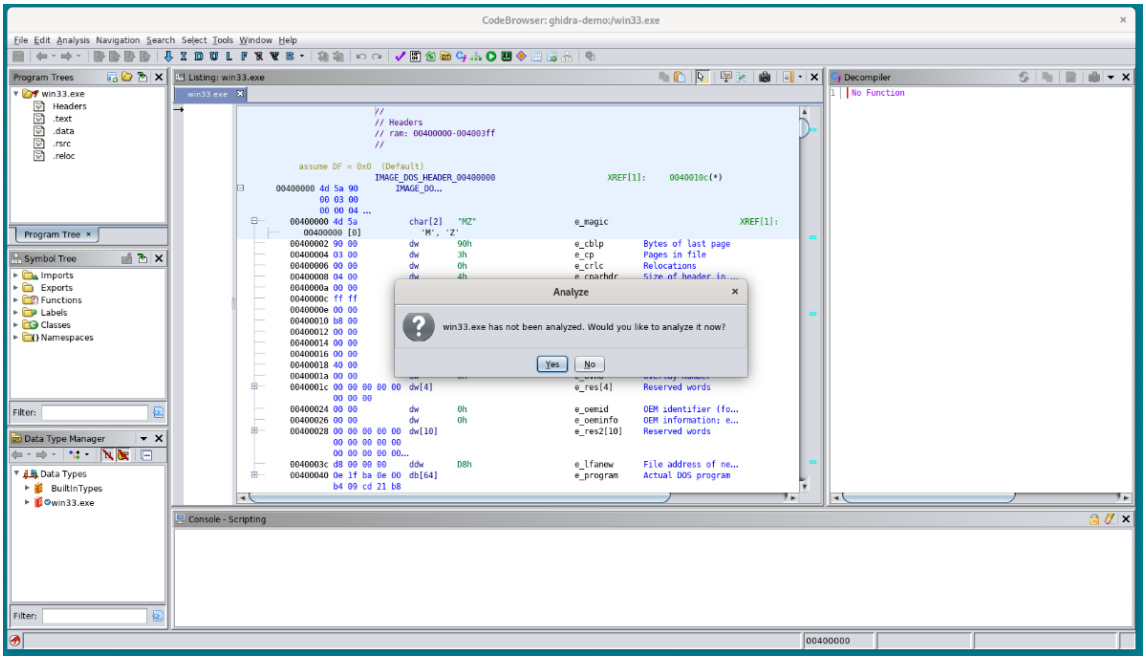
UPOZORENJE! Zbog rizika od zaraze računala, ne preporučujemo preuzimanje i rad s pravim zlonamjernim softverom bez odgovarajućeg predznanja, prethodnog iskustva i mjera opreza. Ako se ipak odlučite pratiti korake u dokumentu i samostalno analizirati softver, obavezno to činite isključivo u izoliranom virtualnom računalu. Pritom virtualno računalo ne smije biti spojeno na internet, a smije imati samo jedan dijeljeni *read-only* direktorij.

Nakon što ste uspješno uvezli program, možete ga otvoriti alatima dostupnima u Ghidri. Standardni alati su „*CodeBrowser*" (ilustriran glavom zelenog zmaja) i „*Version Tracking*" (ilustriran otiscima koraka). Program se u alatu otvara povlačenjem i ispuštanjem učitanog programa nad alatom.



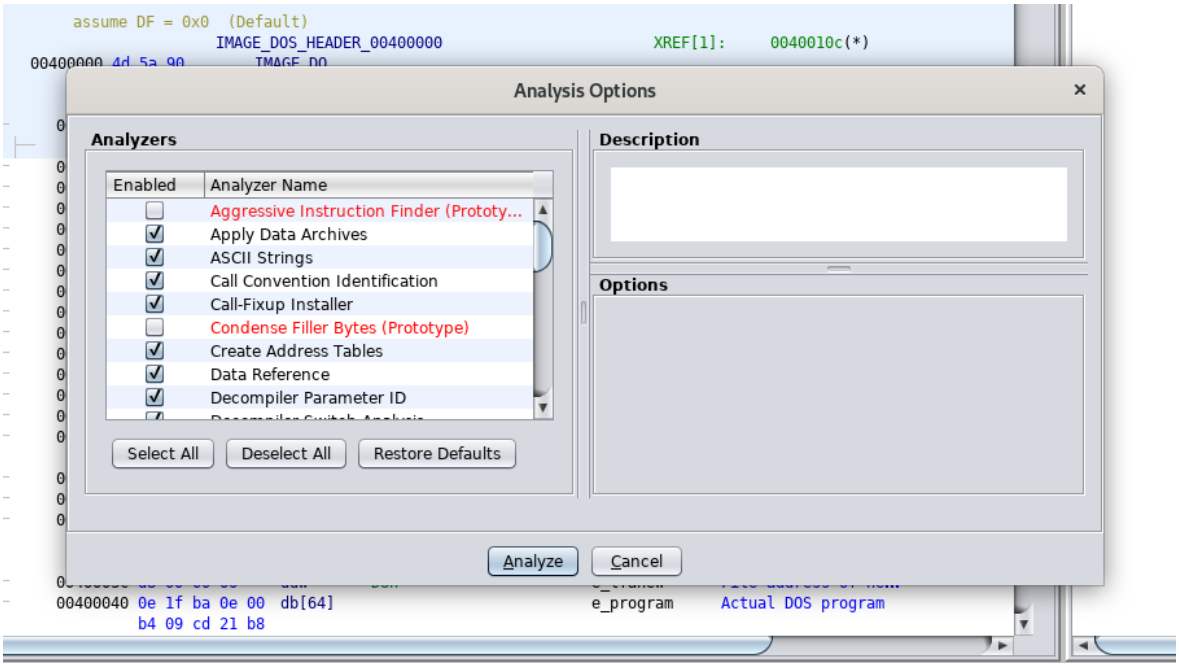
Slika 11 *Otvaranje programa u alatu „CodeBrowser"*

U nastavku će se demonstrirati korištenje alata „*CodeBrowser*" za pretraživanje koda te ćemo zato naš zlonamjerni program povući i ispustiti na ikonu zelenog zmaja.

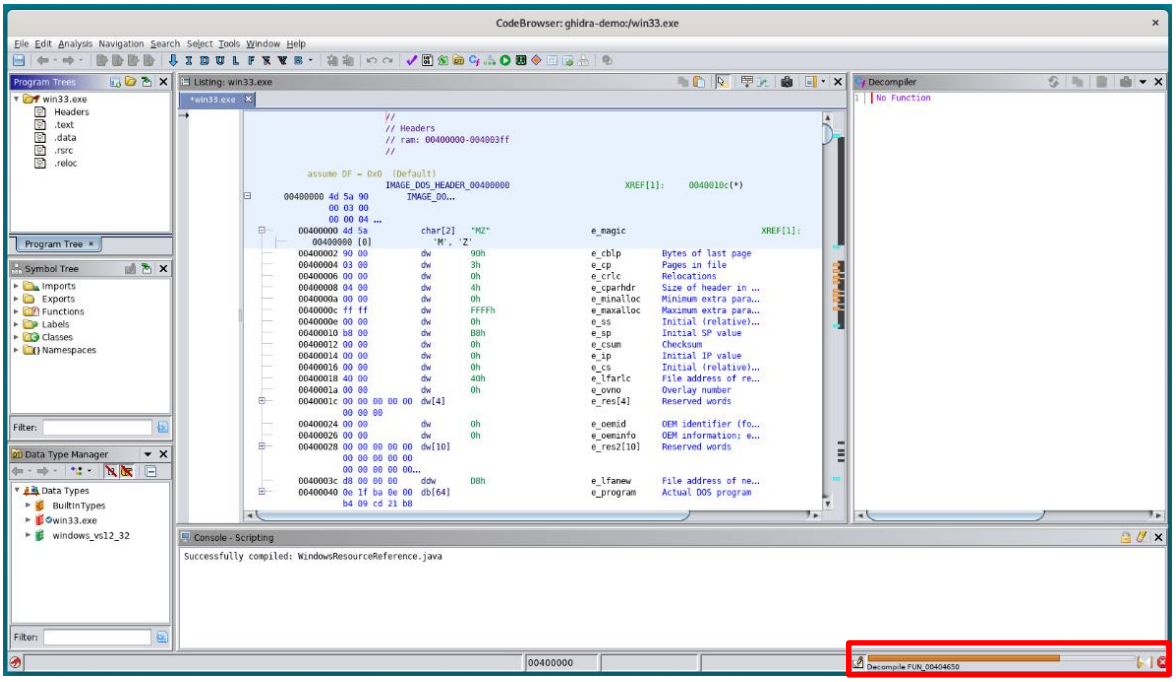


Slika 12 *Analizu programa alatom „CodeBrowser"*

Kada prvi put otvorimo program u „*CodeBrowseru*“, Ghidra će nam ponuditi da ga analizira. Kliknite „*Yes*” te će vam u sljedećem prozoru biti ponuđene razne opcije. Uz svaku opciju navedeno je i kratko objašnjenje. Npr. opcija „*ASCII Strings*“ traži sve nizove ASCII znakova minimalno određene duljine. Nakon što odaberete sve opcije koje želite kliknite „*Analyze*”. Možete ostaviti izvorno zadane postavke.



U donjem desnom kutu možete pratiti tijek analize. Analiza može potrajati nekoliko minuta, ili čak i više, za velike programe.

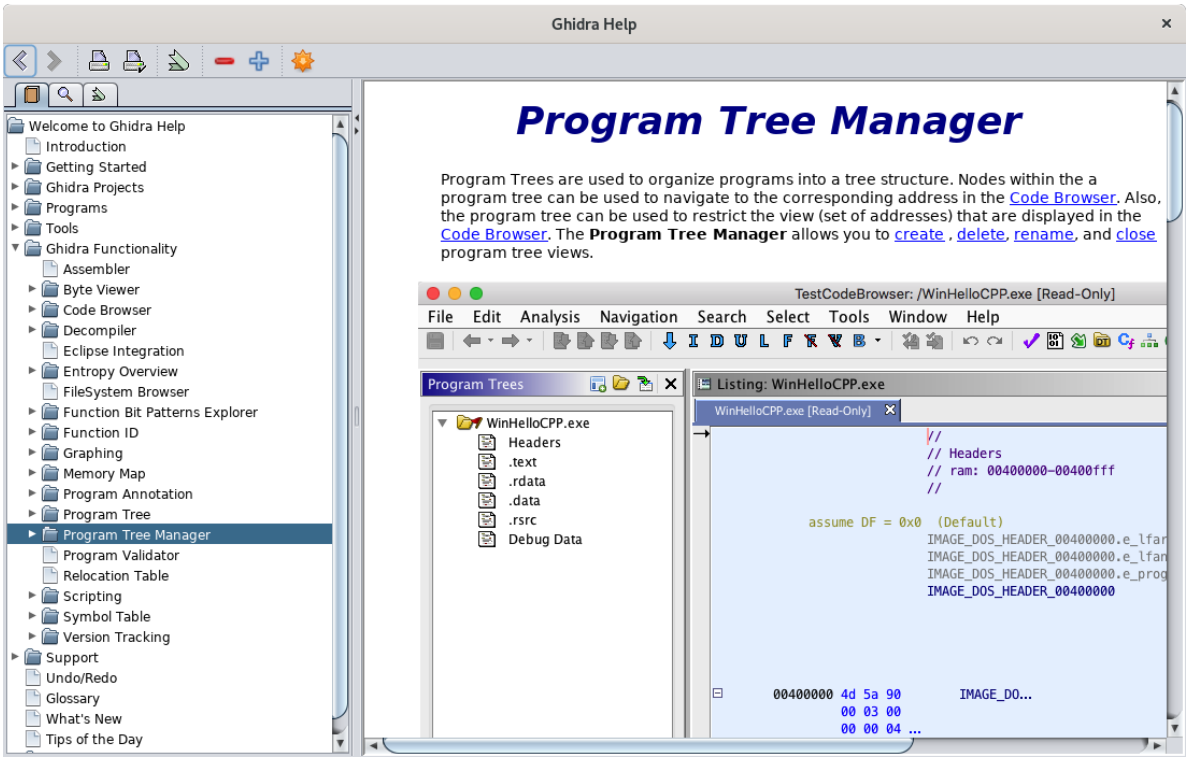


Slika 13 *Analiziranje programa alatom „CodeBrowser“*

Najvjerojatnije ćete unutar alata „*CodeBrowser*“ provesti najviše vremena pa je korisno upoznati se s njegovim funkcionalnostima. „*CodeBrowser*“ se sastoji od više prozora: „*Program Trees*”, „*Symbol Tree*”, „*Data Type Manager*”, „*Listing*”, „*Decompiler*”, „*Console*”.

- „*Program Trees*“ (hrv. stabla programa), prikazuje strukturu programa i omogućava jednostavnu navigaciju kroz nju.
- „*Symbol Tree*“ (hrv. stablo simbola) koristi se za pronalazak simbola i navigaciju.
- „*Data Type Manager*“ (hrv. voditelj struktura podataka) koristi se za pronalazak, primjenu i stvaranje struktura podataka.
- „*Listing*“ (hrv. ispis), ili glavni prozor u sredini, prikazuje asemblerske instrukcije analiziranog programa. Unutar ovog prozora možete pritisnuti tipku „G” na tipkovnici da otvorite „*Go To...*” prozor.
- „*Decompiler*“ (hrv. programski prevoditelj u viši jezik) pokušava prikazati asemblerske instrukcije kao izvorni C kod. Ovaj prozor je usko povezan s prozorom „*Listing*“ te će podcrtavanje instrukcija/koda u jednom podcrtati iste i u drugom.
- „*Console*“ (hrv. konzola) prikazuje stanje i izlaze skripta unutar Ghidre.

Ghidra ima odličnu dokumentaciju te se za bilo koji prozor, izbornik ili radnju pritiskom na tipku „F1” na tipkovnici prikazuje relevantna pomoćna dokumentacija.

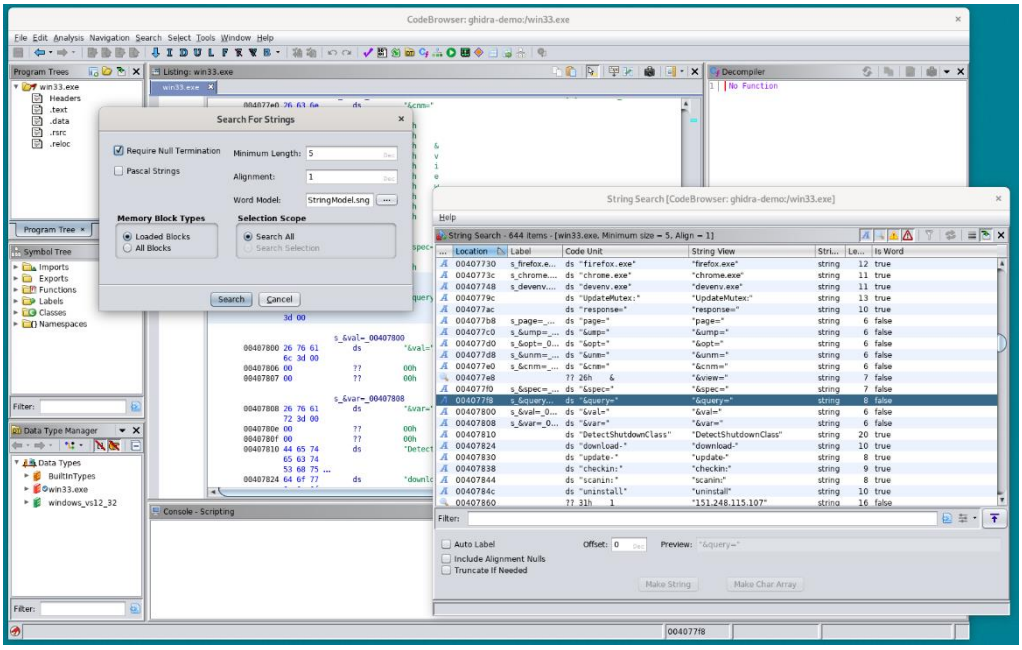


Slika 14 *Prozor s objašnjenjima*

Isto tako, vrlo koristan je i službeni šalabahter.

3.2 Statička analiza programa u Ghidri

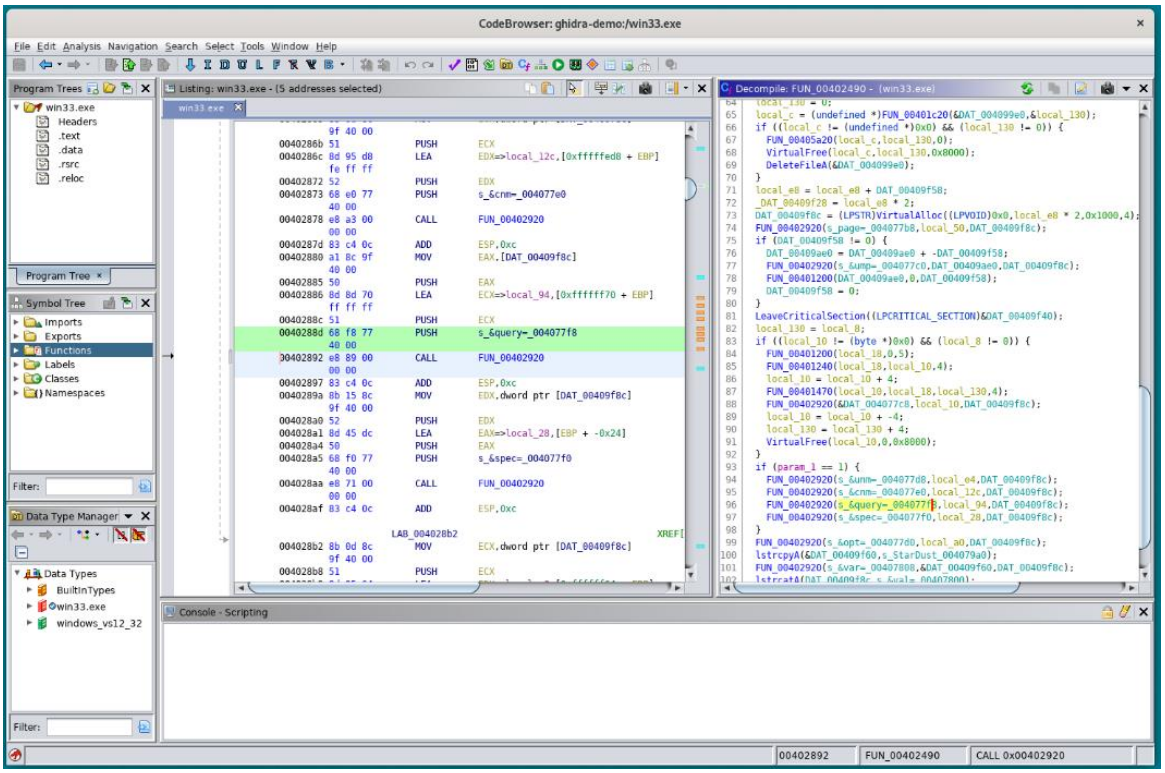
Sad kad smo prošli osnove Ghidre, možemo započeti analizu našeg programa. Započet ćemo našu analizu pretraživanjem znakovnih nizova (engl. *strings*) unutar analiziranog programa. Ovo je dobra početna točka ako analizirani program ne koristi nikakve metode pakiranja/kriptiranja. Znakovni nizovi lako se mogu pretražiti klikom na „*Search->For strings...*". Otvorit će se prozor u kojem se mogu podesiti parametri pretrage, ali u ovom slučaju to nije potrebno nego je dovoljno samo kliknuti na „*Search*".



Slika 15 *Pretraživanje znakovnih nizova*

Ovdje, među znakovnim nizovima, vidimo znakovni niz: „&query=" na adresi 0x004077f8. Taj znakovni niz nas zanima jer izgleda kao dio HTTP zahtjeva koji bi se možda mogao koristiti za komunikaciju s naredbenim i kontrolnim poslužiteljem (engl. *command & control server* ili *C&C server*) koji je pod napadačevom kontrolom.

Nakon pronalaska tog znakovnog niza zanima nas gdje se sve on koristi - tražimo tzv. „*cross-reference*". Pretpostavljamo da bi nam ta funkcija mogla pružiti neke korisne informacije. Sve reference pronalazimo desnim klikom na naš znakovni niz „*References > Show References To Address*". Pronalazimo jednu referencu na adresi 0x0040288d.

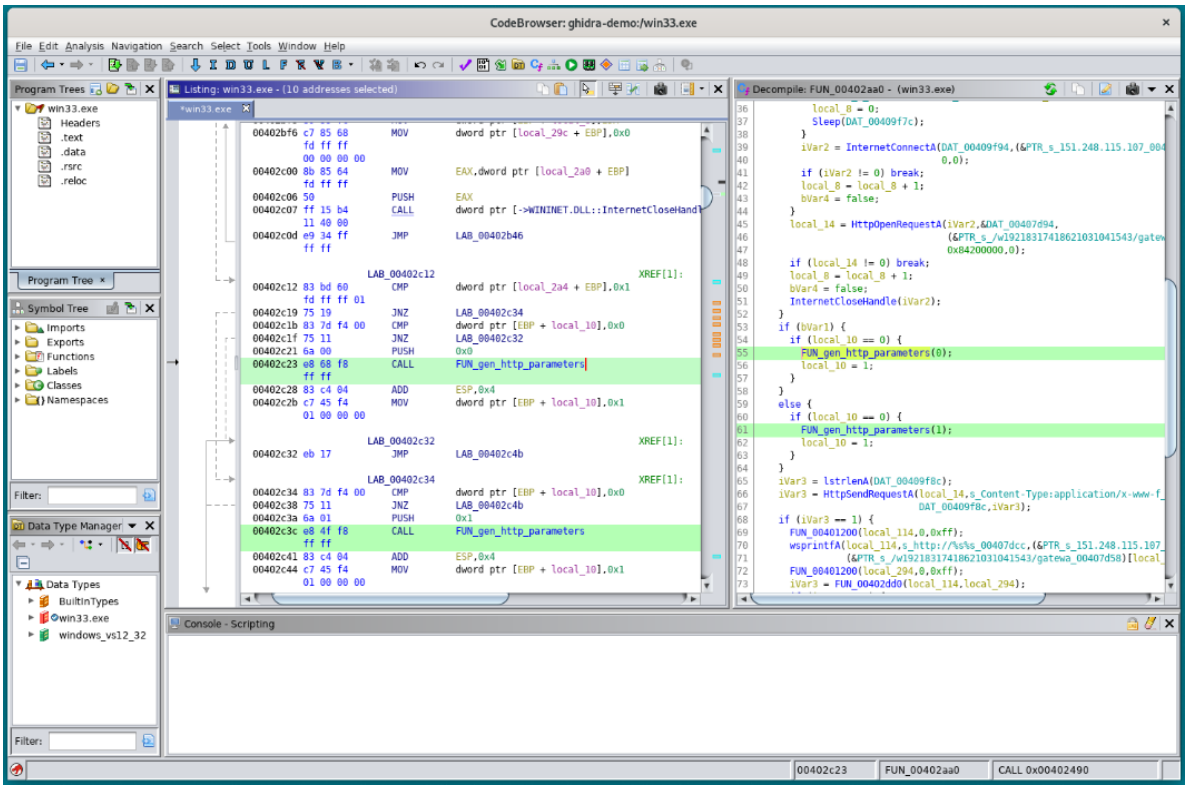


Slika 16 ***Referenca na znakovni niz „&query=”***

Sudeći po okolnim znakovnim nizovima koji su vidljivi u kodu, možemo pretpostaviti da se tu slažu parametri HTTP zahtjeva koji se vjerojatno koristi za komunikaciju s C&C serverom. Vidimo da je to sve dio jedne funkcije koja počinje na adresi 0x00402490. Početak funkcije najlakše se pronađe koristeći prozor decompilera, pomakom do vrha koda i klikom na deklaraciju funkcije.

Ghidra je tu funkciju nazvala „FUN_00402490” po adresi na kojoj se nalazi. Ovdje je jako korisno preimenovati funkciju u nešto što ćemo kasnije lako prepoznati u kodu, kao npr. „FUN_gen_http_parameters”. To lako napravimo desnim klikom na ime funkcije pa „*Rename Function*”, ili pritiskom na tipku „L” na tipkovnici. Ghidra će sada za nas sve reference na tu funkciju preimenovati.

Istim postupkom kojim smo pronašli gdje se koristi naš znakovni niz, tražimo reference na ovu funkciju. Pronalazimo dvije cross-reference na adresama 0x00402c23, odnosno 0x00402c3c.



Slika 17 ***Reference na funkciju „FUN_gen_http_parameters"***

Sad kad smo interpretirali što funkcija radi, možemo ju preimenovati u korisno ime, kao u npr. „FUN_gen_and_send_http". Ovim postupkom gdje analiziramo funkciju, tražimo njezine *cross-reference*, pa analiziramo funkcije koje ju pozivaju postepeno statički analiziramo naš program. Upravo tako i u praksi često izgleda veći dio statičke analize zloćudnih programa.

4 Zaključak

Uobičajeno je prvi korak određivanja je li neki program zlonamjeran ili ne korištenje nekog antivirusnog softvera koji u većini slučajeva dolazi instaliran skupa s operacijskim sustavom. Ako se sumnja na odluku antivirusnog softvera ili se želi detaljnije pogledati na koji način zlonamjerni program pokušava naštetiti računalu, koriste se besplatni i brzi online alati za automatsku statičku i dinamičku analizu poput VirusTotala i HybridAnalysisa.

Najniža stepenica, tj. najdetaljnija analiza postiže se reverznim inženjerstvom. Iako je najdetaljnija i najviše može reći o ponašanju programa, analiza reverznim inženjerstvom zahtijeva veliko stručno znanje, koncentraciju i više vremena (što naravno ovisi i o iskustvu analitičara).

Alati koji pomažu u reverznom inženjerstvu nastoje olakšati analizu raznim dodatnim funkcionalnostima. Svoje mjesto među takvim alatima pronašla je i Ghidra – moćan, koristan alat koji je uz to besplatan i otvorenog koda. Iako je jedan od najnovijih alata, iznimno je prihvaćen od strane sigurnosnih stručnjaka i stoji uz bok s ostalim alatima koji su već neko vrijeme na tržištu poput IDA-e Pro.