УНИВЕРЗИТЕТ У НОВОМ САДУ

ФАКУЛТЕТ ТЕХНИЧКИХ НАУКА У НОВОМ САДУ

Dokumentacija  
[Sigurnost i bezbednost elektroenergetskog softvera](http://esi.ftn.uns.ac.rs/index.php/predmeti/125-sbes)

PhishingToolkit

Profesor:

Prof. Dr. Imre Lendak

Studenti:

Nikola Lugumerski PR75-2017

Bogdan Vilimonović PR78-2017

**Sadržaj**

[​ 1. Uvod 2](#__RefHeading___Toc535_2553567310)

[​ 2. Dizajn 4](#__RefHeading___Toc537_2553567310)

[2.1 Commands 5](#__RefHeading___Toc539_2553567310)

[​ 2.1.1. Povezivanje komandi i property-a modela unutar MVVM WPF aplikacije 5](#__RefHeading___Toc541_2553567310)

[​ 2.1.2. Komande korišćene u aplikaciji: 7](#__RefHeading___Toc543_2553567310)

[2.2 Libraries 14](#__RefHeading___Toc545_2553567310)

[​ 3. Korišćenje 15](#__RefHeading___Toc547_2553567310)

[3.1 Konfiguracija 18](#__RefHeading___Toc549_2553567310)

# 1. Uvod

Phishing predstavlja maliciozan pokušaj kradje senzitivnih podataka, kao što su korisničko ime, lozinka, podaci sa kreditne kartice, pomoću lažiranih web stranica, kojima korisnik veruje. Primarno se izvode pomoću *email*-ova, kao najautomatizovaniji način izvodjenja napada, ali je i nezanemarujuće veliki broj napada koji se izvode putem telefona, kao i pomoću *instant messaging* aplikacija. Može se koristiti u kombinaciji sa drugim vrstama napada, tako što se zaraženi fajl pošalje u vidu *attachment-a*.



*Slika 1.1 Napad na Informatiku u Novom Sadu koristeći Pwndlocker Ransomware ubačenog pomoću phishing napada*

***Bulk Phishing*** predstavlja masovni, automatizovani napad, koji se oslanja na odredjenu procentažu obmanutih žrtvi. Uspešnost samog napada zavisi od mnogo faktora, ali te faktore vezuje jedna glavna ideja, uspešno oponašanje neke validne organizacije, bilo da je to *Google*, banka, internet provajder, ili bilo koja druga vrsta servisa. Format samog *email*-a bi trebalo biti obaveštenje koje zahteva hitan unos podataka, bilo da je to prijavljivanje „sumnjive aktivnosti na nalogu“, promena korisničke lozinke, aktivacija naloga; izgled bi trebao biti isti ili sličan originalu, jer se računa da žrtva neće obraćati pažnju na detalje. Link bi trebalo zameniti sa sopstvenim malicioznim linkom koji vodi ka prijavnoj formi, koja bi takodje poželjno trebala izgledati kao i prijavna forma organizacije umesto koje smo poslali *email*.

***Spear Phishing***predstavlja suprotnost *Bulk Phishing-a* pri kojem se napad koncentriše na odredjene individue, koristeći personalne podatke pri sastavljanju *email*-a, kako bi napadač oponašao nadredjene ili prijatelje tražeći osetljive podatke, kao što su intelektualna svojina, računi, korisnička imena, lozinke, pristup osetljivim podacima. Prvo naučno istraživanje na temu Socijalnog *Phishing-a*, pri kome se koriste javni podaci sa socijalnih mreža, koji koristi prijateljstva, dalo je pozitivne rezultate u 70 posto slučajeva. Unutar organizacija napadi su okrenuti ka posebnim departmanima, kako bi na primer iz finansijskog uzeli novčane izveštaje, ili kako bi preko IT departmana dobili pristup mreži.

***Whaling***predstavlja suženi *Spear Phishing* pri kome se napadač fokusira samo na visoke članove korporativne hijararhije, kao što su direktori, menadžeri i upravni odbori. U ovom slučaju su *email*-ovi visoko personalizovani, kako bi procentaža uspešnosti bila što veća. Mogu biti u formi legalnih dokumenata, zahteva podredjenih, žalbi.

# 2. Dizajn

*Slika 2.1 Dizajn Projekta*

PhishingApp predstavlja glavni deo aplikacije pomoću kojeg obavljamo sastavljanje, slanje *phishing* *email*-ova kao i posmatranje statistika, kao i prikazivanje korisnika koji su uneli svoje podatke u web formu. Sastavljen je pomoću MVVM programske paradigme u WPF framework-u, čime se poštuju SOLID principi programiranja. MVVM model predstavlja najbolji način za razvijanje WPF aplikacija, zbog lakšeg održavanja koda, njegove čitljivosti, kao i mogućnosti ponovne upotrebe već kreiranih komandi. Svako od polja se vezuje za posebni *Command* koji izvršava odredjenu funkcionalnost kada u programu dodje do nekog dogadjaja.

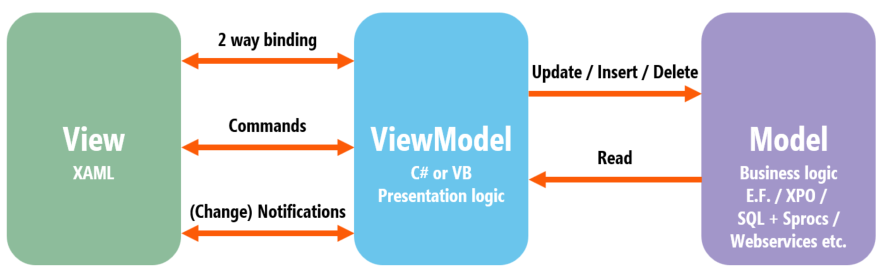
WebsiteDummy predstavlja demo website-a prekopiranog sa FTN Canvas platforme, koji radi na lokalnoj mašini, i pri prijavi izbacuje poruku kako je pogrešno korisničko ime ili lozinka.

Common predstavlja Class library pomoću koga WebsiteDummy šalje korisničko ime i lozinku kao i vreme prijave, i te informacije se upisuju u bazu PhishingApp-a.

## 2.1 Commands

Komande predstavljaju uniforman način za povezivanje button-a UI-a sa traženom funkcionalnošću, kao i to da komande podržavaju „binding“ podataka( property-a iz Modela). Komande nisu direktno povezane sa njihovim pozivaocom, pa ista komanda može da se koristi na više mesta (menu item, toolbar button, keyboard...). Omogućavaju i validaciju kroz CanExecute() metodu, kao jednostavan način za ograničavanje korisnikovih aktivnosti u smislu usmeravanja ka pravilnom korišćenju programa.

### 2.1.1. Povezivanje komandi i property-a modela unutar MVVM WPF aplikacije



Svaki textbox koji se koristi na UI-u je preko XAML koda koji se nalazi iza design-a povezan sa odredjenim property-em koji se nalazi u Modelu.

Veza može biti: *OneWay, TwoWay, OneWayToSource,* koristimo onu koja nam je potrebna u datom trenutku.

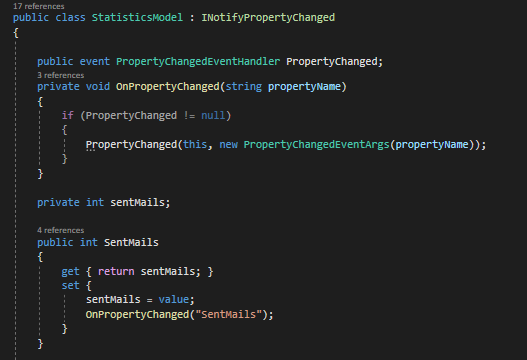
Primer povezivanja koda na View strani je prikazano na sledećoj slici (Slika 2.1.1 Povezivanje na View-u)



*Slika 2.1.1 Povezivanje na View-u*

A ako bismo želeli da svaki put prilikom ažuriranja samog property-a modela dođe i do ažuriranja sadržaja u textbox-u moramo dodati i tag UpdateSourceTrigger=PropertyChanged.

Kako bi ovaj tag funkcionisao, sama klasa u Modelu mora da nasledjuje INotifyPropertyChanged, i da implementira sam interfejs na validan način(Slika 2.1.2. Nasledjivanje i implementacija interfejsa InotifyPropertyChanged)



*Slika 2.1.2. Nasledjivanje i implementacija interfejsa INotifyPropertyChanged*

**ViewModel** povezuje View sa Modelom, gde podatke predaje UI-u i pruža funkcionalnost kroz Komande. U suštini on instancira sve objekte iz Modela koji su nam potrebni, kao i Komande.

Shodno ovom načinu povezivanja textbox-a i property-a na sličan način se i vezuju komande za button-e ( Slika 2.1.3. Povezivanje button-sa sa Komandom na View).



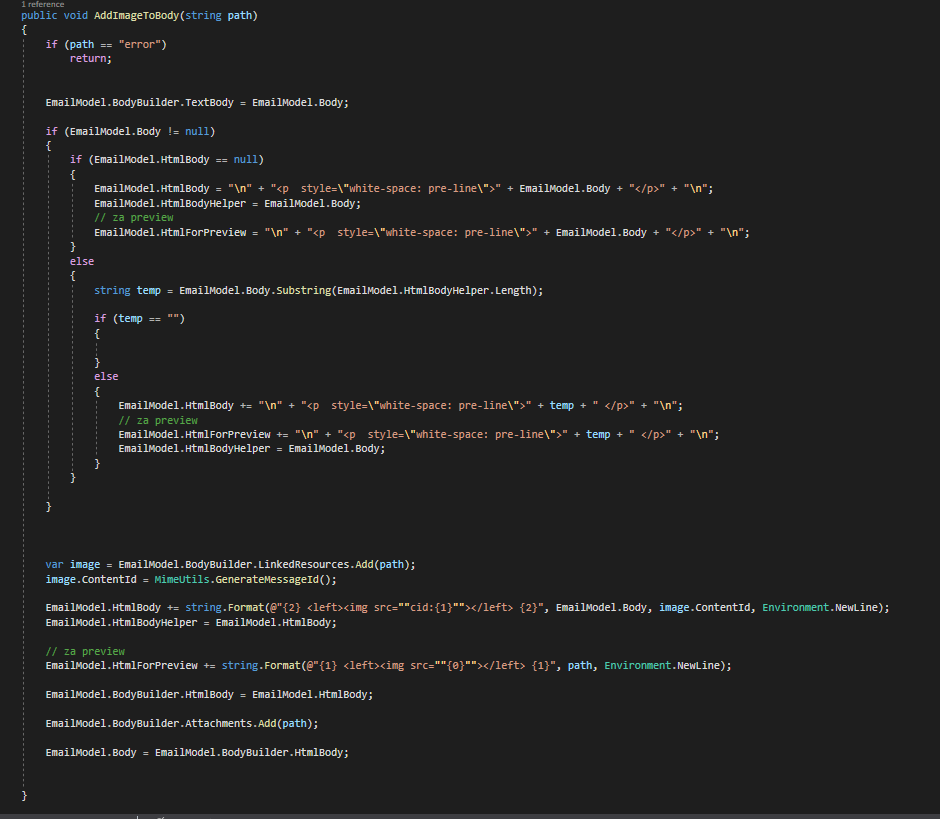
*Slika 2.1.3. Povezivanje button-sa sa Komandom na View*

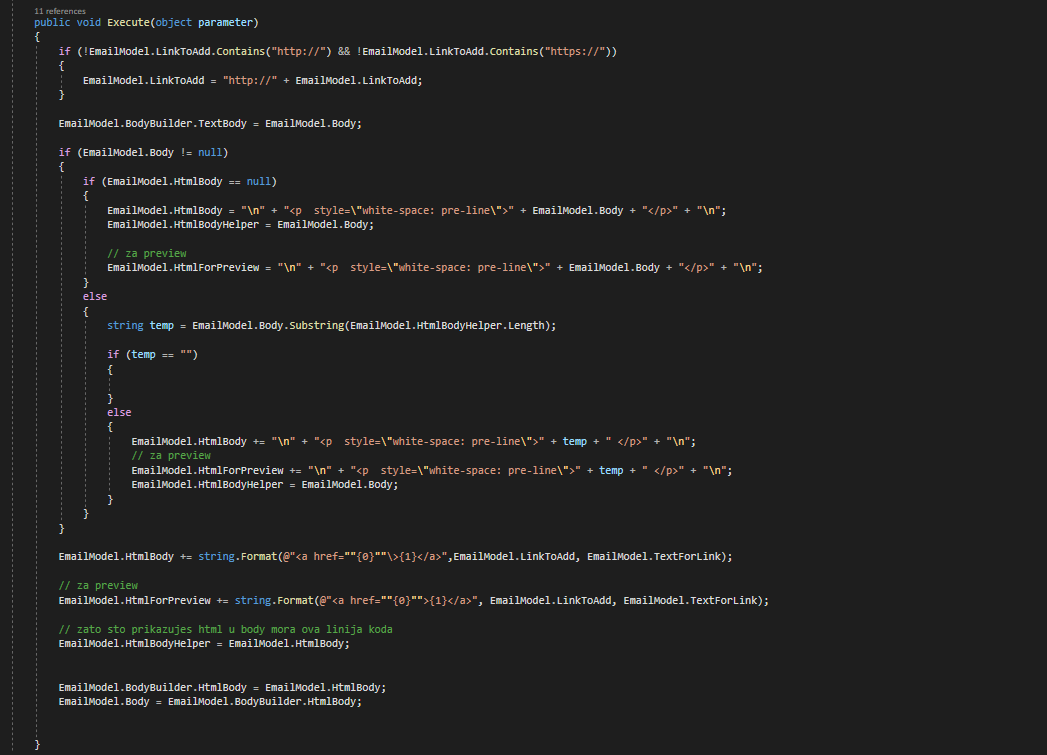
Svaka komanda nasledjuje ICommand interfejs i zahteva da se implementiraju 3 metode:

1. protected void RaiseCanExecuteChanged();
2. public bool CanExecute(object parameter);
3. public void Execute(object parameter);

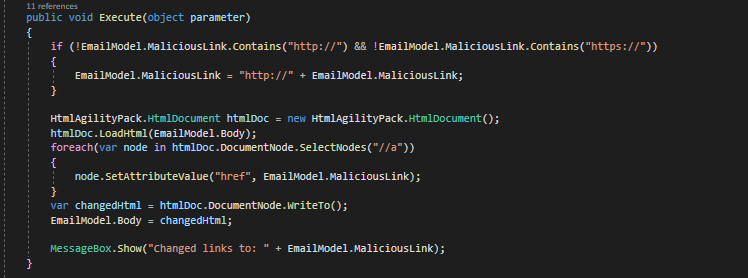
### 2.1.2. Komande korišćene u aplikaciji:

1. AddImageCommand – otvara File Explorer pozivajuci funkciju BrowseImageFiles() i izabranu sliku formatira za email i programski generiše html kod koji je potreban prilikom ubacivanja inline fotografija u email.

*Slika 2.1.3 Generisanje inline image-a u email-u*

1. AddLinkCommand – dodaje link u mail, usput programski generiše html kod i dodaje ga u email.

*Slika 2.1.4 Dodavanje linka unutar email-a*

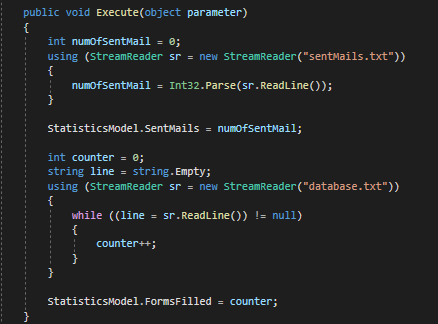
1. ChangeLinksCommand – koristi se u slučaju kad importujemo email i želimo da u sadržaju tog mail-a sve linkove zamenimo našim malicioznim linkom.

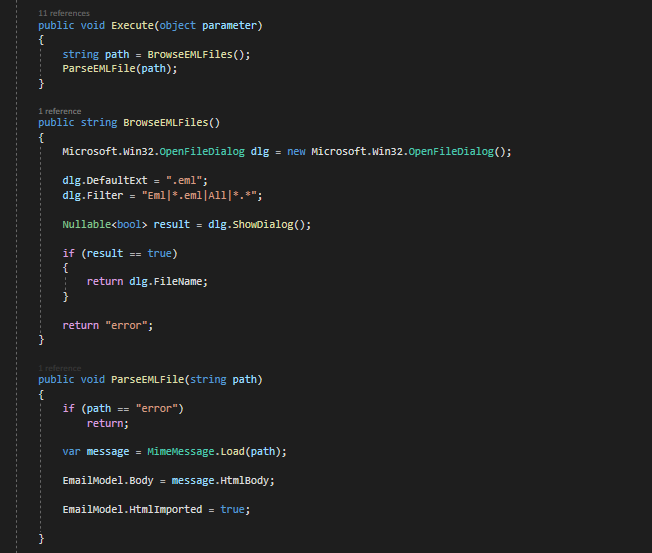
*Slika 2.1.5 Zamena linkova u importovanom mailu sa malicioznim linkom*

1. EmailReadCommand – čita sve mail-ove iz već kreiranog .txt dokumenta i popunjava textbox na UI-u, prilikom čega proverava da li su svi email-ovi validni

 *Slika 2.1.6 Učitavanje mail-ova iz .txt fajla*

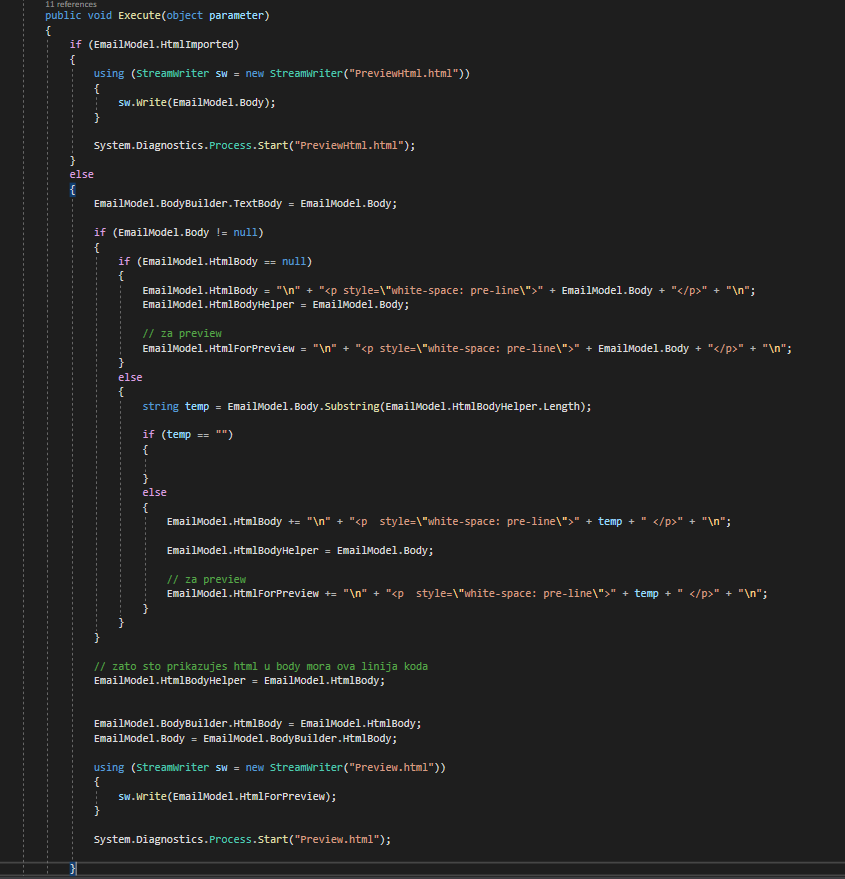
1. InitializeCommand – učitava sačuvano stanje aplikacije prilikom poslednjeg gašenja iste.

*Slika 2.1.7 Učitavanje sačuvanog stanja aplikacije*

1. ParseEmailCommand – pretvara skinuti email u html format pogodan za editovanje i dalje slanje.

*Slika 2.1.8 Učitavanje skinutog email-a i pretvaranje istog u html format*

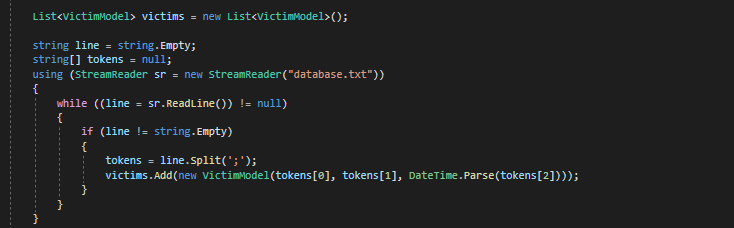
1. PreviewEmailCommand – omogućava da vidimo kako će izgledati izgenerisani mail pre slanja, s tim da postoje dve razlicite situacije, a to je da li je ucitan email ili smo ga sami pravili



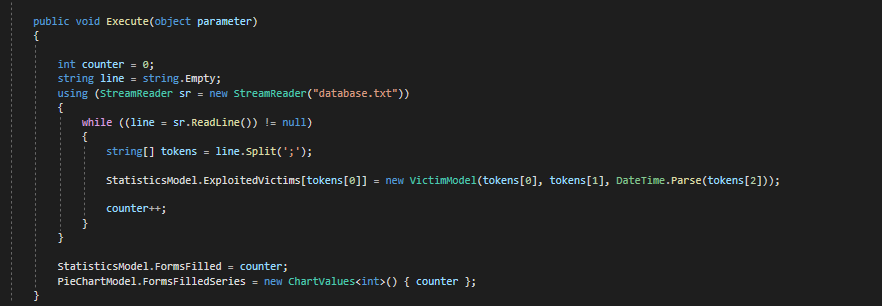
*Slika 2.1.9 Pregled email-a pre slanja*

1. SendEmailCommand – povezuje se sa gmail SMTP-om koristeći već kreiran nalog i šalje phishing poruku na sve učitane mail-ove.

*Slika 2.1.10 Pakovanje email-a i slanje na sve email-ove*

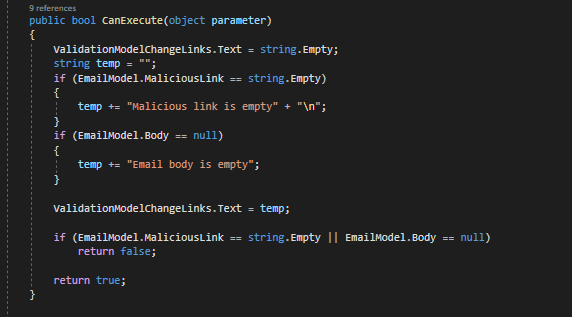
1. ShowExploitedVictimsCommand – u novom prozoru prikazuje sve žrtve koje su popunili formu na mockovanom sajtu i time otkrili podatke.

*Slika 2.1.11. Popunjavanje novog prozora sadrzajem*

1. StatisticCommand – popunjava i osvežava grafik u aplikaciji na svakih 10 sekundi.

*Slika 2.1.12 Osvežavanje vrednosti grafika*

Potrebno je naglasiti da svaka Komanda ima implementiranu CanExecute metodu, koja predstavlja validaciju i popunjava tooltip button-a kako bi se korisniku olakšalo snalaženje u aplikaciji.

*Slika 2.1.13 Primer CanExecute medote iz ChangeLinks Komande*

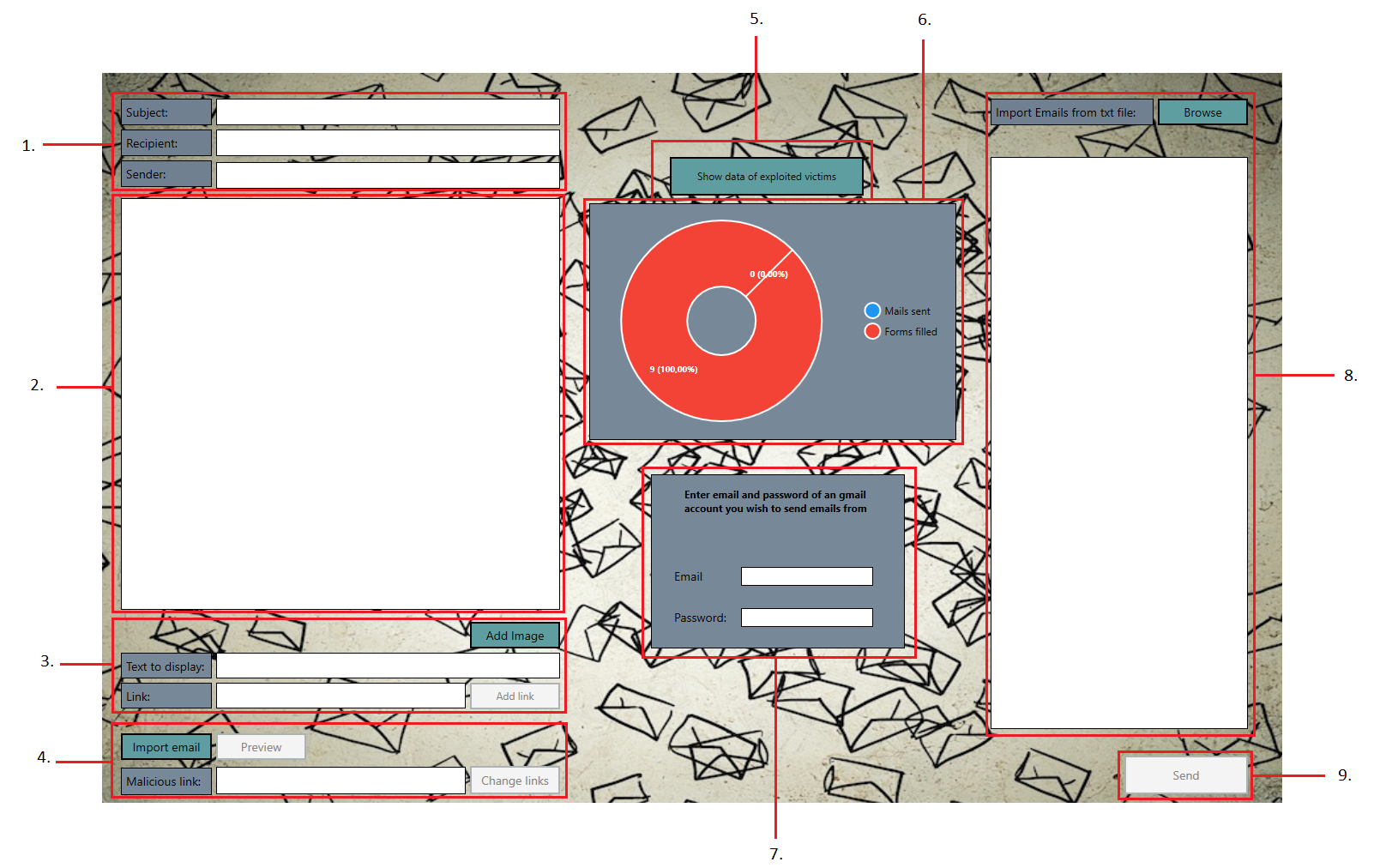
## 2.2 Libraries

HtmlAgilityPack – korišćen prilikom parsiranja tela poruke, kako bi mogli izvući sve „<a>” tagove kao i njihov „href“ attribute, koje menjamo kako bismo zamenili postojeće linkove sa našim malicioznim

MailKit – korisna multipraktična biblioteka koja sadrži logiku koja standardizuje poruku i parsira eml fajlove,kao i sa konekcijske strane, pravi smtp klijenta i uspostavlja vezu sa serverom

LiveCharts – grafička biblioteka korišćena pri prikazivanju statistike odnosa poslatih email poruka i korisnika koji su se prijavili preko malicioznog sajta

# 3. Korišćenje

*Slika 3.1 Izgled PhishingApp prozora*

1. Polja za unos Naslova (Subject), kao i imena Pošiljaoca (Sender) i Primaoca (Recipient). Preporučuje se da obraćanje posebne pažnje na ime Sendera, koje se mora poklapati sa sadržajem samog *email*-a, jer se u Inbox prozoru Gmail-a ne prikazuju same *email* adrese već samo imena Pošiljalaca, tek kada korisnik udje u sam *email*, može se videti da adresa nije ispravna, ali se računa na nepažnju korisnika, i validan izgled, koji zavarava korisnika. Potrebno je uneti neke vrednosti, polja se ograničavaju validacijom pa ne smeju biti prazna.

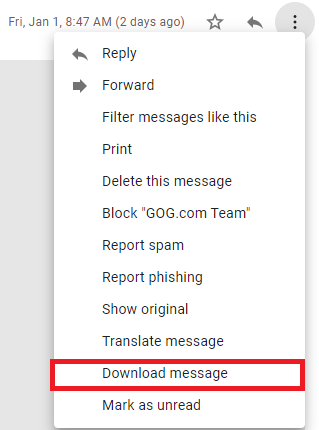
2. Polje za unos glavnog dela poruke (Body). Omogućen je unos regularnog tekstualnog sadržaja, kao i automatsko generisanje html koda prilikom unosa teksta i slika. Potrebno je uneti sadržaj, polje se ograničava validacijom i ne sme biti prazno.

3. Polja koja omogućavaju dodavanje slika i linkova. Pri dodavanju slika tekst koji smo ukucali u glavni deo poruke, dodaje se novi tag za sliku. Pri dodavanju linka, tekst takodje biva prepravljen u html kod, ali uz opciju dodavanja samog linka, imamo i mogućnost postavljanja samog teksta linka.

4. Polja za importovanje eksternog *email*-a, pregledanje sadržaja glavnog dela poruke u pretraživaču, kao i menjanje svih linkova poruke, našim malicioznim linkom.

Kako bi učitali željeni *email*, pristupamo putem pretraživača Gmail sajtu. Ulazimo u *email* koji želimo skinuti, i u gornjem desnom uglu selektujemo tri tačke, i biramo opciju „Download Message“ (Slika 3.2). U našem programu selektujemo skinuti fajl klikom na dugme *Import email*, koji ima ekstenziju .eml. Fajl se učitava, i dobijemo izgenerisani html kod poruke, u body-u poruke. Ukoliko vizuelno želimo videti izgled *email*-a selektujemo opciju *Preview*, njom se u podrazumevanom pretraživaču prikazuje izgled poruke.

Pri zameni linkova unosimo link u polje sa nazivom *Malicious link* i selektujemo change links, ukoliko nije unet prefiks „http://“ on biva dodat automatski. Ukoliko želimo da isprobamo naš *WebsiteDummy*, nakon što pokrenemo projekat, iz pretraživača kopiramo link i unosimo ga u polje.

*Slika 3.2 Preuzimanje Email-a sa Gmail Sajta*

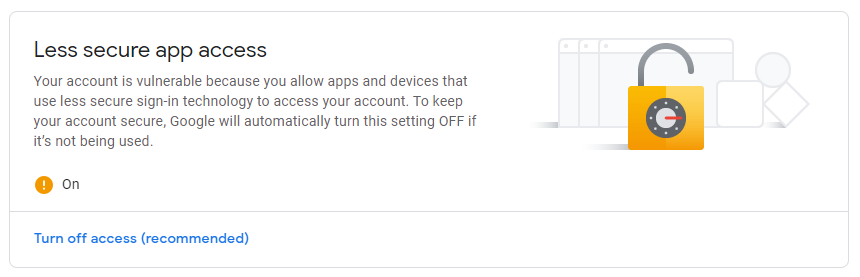
5. Dugme za prikaz korisničkih imena i lozinki korisnika koji su uneli svoje podatke putem *WebsiteDummy*-a. Otvara novi prozor u kojem se prikazuju u tabeli uneti podaci i vreme njihovog unosa.

6. Grafički prikaz statistike odnosa poslatih *email*-ova i količine korisnika koji su popunili formu.

7. Pre nego što počnemo objašnjavati kako se koristi nalog za slanje kreiranih phishing *email-ova*, treba obratiti posebnu pažnju na samo ime naloga kao i izgled samog imena mail-a. Ono treba da bude što verodostojnije i sličnije onome sa kojeg korisnici očekuju da prime baš takav mail. Na primer, u slučaju da želite da se predstavljate kao Otp Banka Srbije, vaš novokreirani nalog bi trebao da ima oblik poput „otpbankasrbije@gmail.com“ ili slično (u ovom ne bi imalo smisla slati mail sa naloga koje izgleda kao npr. [petartestovic@gmail.com](mailto:petartestovic@gmail.com)).

Predstavlja polje za unos napravljenog Gmail naloga. U slučaju neuspešne autentifikacije izbacuje se poruka, iznad polja za unos. Po podrazumevanim opcijama Google blokira slanje *email*-ova sa nepoznatih izvora (aplikacija). Kako bi omogućili slanje, ulogujemo se na naš nalog i onda pratimo jedan od dva načina:

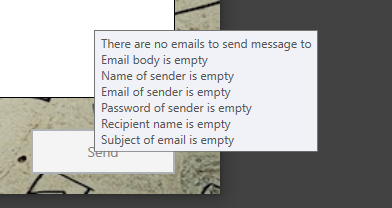
* Pristup putem linka https://myaccount.google.com/lesssecureapps
* Ulazimo na Gmail, selektujemo u gornjem desnom uglu krug sa prvim slovom imena našeg naloga, i kliknemo na „Manage your Google Account“, nakon toga sa levog menija selektujemo Security, i u njemu nalazimo opciju za manje sigurne uredjaje (Slika 3.3), selektujemo je i stavljamo na „ON“

*Slika 3.3 Sigurnosni Prozor Google Account*

Potrebno je uneti vrednosti u polja, inače je validacijom onemogućeno slanje.

8. Polje za učitavanje mejlova, kao i njihovo prikazivanje. Sadrži dugme *Browse* koje nam otvora *file explorer* u kome nalazimo fajl sa .txt ekstenzijom, u kojima su nam napisani email adrese korisnika kojima ćemo poslati email; fajl mora imati format tako da email adrese budu jedna ispod druge. Na polju postoji validacija i kako bi omogućili slanje moramo uneti bar jednu email adresu. Adrese moraju biti u pravilnom formatu, kako bi im poruka bila poslata.

9. Dugme za slanje koje je onemogućeno u slučaju da nisu popunjena sva polja koja se validiraju. U slučaju da niste sigurni zašto vam je onemogućeno da pritisnete odredjeni button, jednostavno postavite miš iznad button-a i pojaviće vam se tekst ( Slika 3.4 Tooltip) šta niste ispunili od zahteva da bi button mogao biti aktivan.



*Slika 3.4 Tooltip*

## 3.1 Konfiguracija

Preporučuje se korišćenje sopstvenog smtp servera, jer to donosi dva važna napretka:

* Nezavisnost od Google servera – u svakom trenutku mogu primetiti sumnjivu povećanu aktivnost i blokirati naše poruke
* Potencijalni „*Email Spoofing*“ – lažiranje email adrese pošiljaoca pri čemu upotpunjujemo iluziju validnosti *email* poruke

Preporučeno je korišćenje hMailServer-a, jer nudi relativno lak proces instalacije, kao i podešavanja, koji se mogu naći na sledećem linku:

https://medium.com/@coffmans/setup-your-own-simple-smtp-server-how-to-c9159cfc7934

Jedan broj ISP-a (Internet Provajdera) blokira saobraćaj na port-u 25 koji ovaj program koristi za komunikaciju, kako bi proverili da li vaš provajder dozvoljava ovo pristupite sledećem linku <http://portquiz.net:25/>. Ukoliko vam je port zablokiran, sajt neće biti dostupan.

Još jednu opciju predstavlja iznajmljivanje smtp servera, medju kojima su popularni servisi SMTP2GO, Sendinblue, Amazon SES…

Kada smo napravili ili iznajmili željeni server, parametre kao što su hostname, port i omogućavanje SSL popunjavamo u okviru App.config fajla PhishingApp projekta. Kredencijale za prijavljivanje/autentifikaciju popunjavamo u okviru samog programa (polja označena sa 7. u prethodnom poglavlju).

Kako bi u potpunosti iskoristili funkcionalnosti datog projekta, potrebno je zakupiti domen i pokrenuti na web serveru dati WebsiteDummy, ili drugu malicioznu web stranicu.