

Forum:[Hack Forums](#)

Thread: [Python 3.* Email Spammer](#)

FuzzOG - May 10, 2017, 10:26 pm

THIS PROJECT IS FREE!

This is a very simple python program (30 lins of code), that allows you to spam any email address. Hope you guys enjoy!

To login with "your" email you need to allow for less secure apps to connect to your google account (myaccount.google.com/lesssecureapps).

Source Code: ghostbin.com/paste/94k68]

saliva - May 10, 2017, 10:35 pm

You don't catch exceptions, and `smtplib.sendmail()` can raise one. This will give the illusion of working even if it doesn't.

FuzzOG - May 10, 2017, 11:06 pm

(May 10, 2017 10:35 PM)saliva Wrote: [\[CITE\]](#) You don't catch exceptions, and `smtplib.sendmail()` can raise one. This will give the illusion of working even if it doesn't. I have tested this and has worked. I'm not sure if every single email sends, but works alright. Ill definitely look into improving it!

Mohinder - May 11, 2017, 12:47 am

(May 10, 2017 10:26 PM)FuzzOG Wrote: [\[CITE\]](#) THIS PROJECT IS FREE!

This is a very simple python program (30 lins of code), that allows you to spam any email address. Hope you guys enjoy!

To login with "your" email you need to allow for less secure apps to connect to your google account (myaccount.google.com/lesssecureapps).

Source Code: ghostbin.com/paste/94k68]

Since Gmail doesn't allow spamming this thing won't actually spam. I might be wrong here but Google has an limit of 500 emails per day. If you want a cool spam script, I got one for you. Able to spam over few 100 mails every second exploiting other websites.

FuzzOG - May 11, 2017, 1:12 am

(May 11, 2017 12:47 AM)Mohinder Wrote: [\[CITE\]](#) Since Gmail doesn't allow spamming this thing won't actually spam. I might be wrong here but Google has an limit of 500 emails per day. If you want a cool spam script, I got one for you. Able to spam over few 100 mails every second exploiting other websites. Yo, can you hit me up with a PM, if your willing to show me the script

Malicious Intent - May 11, 2017, 1:16 am

(May 11, 2017 12:47 AM)Mohinder Wrote: [\[CITE\]](#) Since Gmail doesn't allow spamming this thing won't actually spam. I might be wrong here but Google has an limit of 500 emails per day. If you want a cool spam script, I got one for you. Able to spam over few 100 mails every second exploiting other websites.

Thats correct.. I think they only let you send 100 emails at the time and 400-500 per day total.

BungeeGum - May 11, 2017, 7:06 am

Wow! Your topic here is very interesting, might giving it a try. ;)

FuzzOG - May 11, 2017, 12:00 pm

(May 11, 2017 07:06 AM)BungeeGum Wrote: [\[CITE\]](#) Wow! Your topic here is very interesting, might giving it a try. ;)IF you find ways to improve it please let me know :D

Malicious Intent - May 11, 2017, 12:07 pm

(May 11, 2017 12:00 PM)FuzzOG Wrote: [\[CITE\]](#) IF you find ways to improve it please let me know :D
If you are asking for the ways to improve it... Here is couple of things you can use to make it work with html and also attachments plus specify the date format and all ..

```
Code from os.path import basename
from email.mime.application import MIMEApplication
from email.utils import COMMASPACE, formatdate
```

If you need the functioning code for reference just let me know.

?GC? - May 11, 2017, 12:25 pm

Just a thought, but have you thought about monetizing this and turning it into a project? Not only could the script be used for email bombing, you could purchase a relatively cheap SMTP service to allow unlimited mail for mail bombing, and also since SMS messages [texts] have email to SMS gateways, you could use it to SMS bomb with relative ease.. you could then sell this as a service / package to users on HF.

Edit:

Just a quick explanation, email to SMS works like this as an example:

[Phonenumber here]@[carrierid].co.uk

You send emails, and they are received as SMS by the number.. your script could do this easily. Kinda like the one in my sig -.^

Malicious Intent - May 11, 2017, 12:50 pm

(May 11, 2017 12:25 PM)GameCrack Wrote: [\[CITE\]](#) Just a thought, but have you thought about monetizing this and turning it into a project? Not only could the script be used for email bombing, you could purchase a relatively cheap SMTP service to allow unlimited mail for mail bombing, and also since SMS messages [texts] have email to SMS gateways, you could use it to SMS bomb with relative ease.. you could then sell this as a service / package to users on HF.

Edit:

Just a quick explanation, email to SMS works like this as an example:

[Phonenumber here]@[carrierid].co.uk

You send emails, and they are received as SMS by the number.. your script could do this easily. Kinda like the one in my sig -.^

Email -> sms only works with some carriers .. mostly American tho. Its a good idea nonetheless.

?GC? - May 11, 2017, 1:03 pm

I'm absolutely aware, I just figured it'd be relatively easy to implement it Python wise, could be an interesting project and for him to make a little cash on the side whilst helping out the HF community :)

hunter24957 - May 11, 2017, 1:07 pm

Why do you define x, but never actually use it?

It's a waste of precious memory!

Malicious Intent - May 11, 2017, 1:27 pm

(May 11, 2017 01:07 PM)hunter24957 Wrote: [\[CITE\]](#) Why do you define x, but never actually use it?

It's a waste of precious memory!
lol good spot.

Also if you enable the subject in your code by removing # .. it will not work.. you need to add this line before you attach MIME or subject will remain empty.

```
Code msg['Subject'] = subject
```

This will create an issue of stacking emails. Easy fix tho.

```
Code # mailBomb
```

```
# Python ver: 3.*
```

```
# Made by: FuzzOG
```

```
import smtplib
```

```
from email.mime.multipart import MIMEMultipart
```

```
from email.mime.text import MIMEText
```

```
import string, random
```

```
num = 1
```

```
fromaddr = "" #Your email
```

```
PASSWORD = "" #Your password (https://myaccount.google.com/lesssecureapps)
```

```
#PASSWORD = open('pass.txt', 'r').read() #Use to get password from pass.txt
```

```
toaddr = input("Recievers Email: ") #Recievers email
```

```
sub = input("Subject: ")
```

```
body = input("Message: ") #Message
```

```
server = smtplib.SMTP('smtp.gmail.com', 587)
```

```
server.starttls()
```

```
server.login(fromaddr, PASSWORD)
```

```
while True:
```

```
    msg = MIMEMultipart()
```

```
    msg['From'] = fromaddr
```

```
    msg['To'] = toaddr
```

```
    subject=sub+"".join(random.choice(string.ascii_uppercase + string.digits) for _ in range(5))
```

```
    msg['Subject'] = subject
```

```
    msg.attach(MIMEText(body, 'plain'))
```

```
    text = msg.as_string()
```

```
    server.sendmail(fromaddr, toaddr, text)
```

```
    subject=sub
```

```
    print("Your message has been sent to:", toaddr, " (Press CTR + C to stop) Count:", num)
```

```
    num = num + 1
```

server.quit()As you can see in this case the MIME has to be constructed inside the loop for this to work correctly. I personally prefer for loop for this task with specified number of emails.

Multithreading would be an advantage but then again .. gmail will only allow so many emails to be sent so kinda waste of time.

FuzzOG - May 11, 2017, 9:46 pm

(May 11, 2017 12:25 PM)GameCrack Wrote: [\[CITE\]](#) Just a thought, but have you thought about monetizing this and turning it into a project? Not only could the script be used for email bombing,

you could purchase a relatively cheap SMTP service to allow unlimited mail for mail bombing, and also since SMS messages [texts] have email to SMS gateways, you could use it to SMS bomb with relative ease.. you could then sell this as a service / package to users on HF. Thanks man, I am thinking of making a website where you can use this from but I don't know laws around email spamming so... IDK, might try selling as a product on HF

(May 11, 2017 01:27 PM) Malicious Intent Wrote: [\[CITE\]](#) lol good spot.

Also if you enable the subject in your code by removing # .. it will not work.. you need to add this line before you attach MIME or subject will remain empty... Thanks for the help, I will definitely implement some of these.

Chercher - May 13, 2017, 6:48 am

- 1 for using libraries
- 1 for defining unused variable 'x'
- 1 for utilizing ctrl+c to stop loop instead of actually setting a limit
- 1 for autistic variable name-styles
- 1 for not knowing about gmail's email policy.

i don't think there's anything good with this code.

Malicious Intent - May 13, 2017, 7:05 am

(May 13, 2017 06:48 AM) Chercher Wrote: [\[CITE\]](#) -1 for using libraries

- 1 for defining unused variable 'x'
- 1 for utilizing ctrl+c to stop loop instead of actually setting a limit
- 1 for autistic variable name-styles
- 1 for not knowing about gmail's email policy.

i don't think there's anything good with this code.

Whats wrong with using modules? thats how python works. How is it better to write everything from scratch when there are modules written by someone who can do it way better than you ever could and speed up your development process. smtplib is standard and i don't see why one would like to change anything there (probably fuck it up) when it provides functions for everything you will ever need while working with email. Mime is also standard.. there are other modules designed specifically to work with gmail like yagmail so he actually is working with generic python modules not some third party shit.

Why do you care for his variables names? you can use any variable names you want unless they are standard function names or restricted.

If i remember correctly these are the variable names used in python docs for those modules.

It's clear the op is learning and posted his code for review and clearly stated hes waiting for suggestions to improve it.

Instead of bashing someones work why don't you provide us with better code and point out why it's better... then we can criticize it the way you just did..

Chercher - May 13, 2017, 7:14 am

(May 13, 2017 07:05 AM) Malicious Intent Wrote: [\[CITE\]](#) Whats wrong with using modules? thats how python works. How is it better to write everything from scratch when there are modules written by someone who can do it way better than you ever could and speed up your development process. smtplib is standard and i don't see why one would like to change anything there (probably fuck it up) when it provides functions for everything you will ever need while working with email. Mime is also standard.. there are other modules designed specifically to work with gmail like yagmail so he

actually is working with generic python modules not some third party shit.

Why do you care for his variables names? you can use any variable names you want unless they are standard function names or restricted.

It's clear the op is learning and posted his code for review and clearly stated hes waiting for suggestions to improve it.

Instead of bashing someones work why dont you provide us with better code and point out why its better... then we can criticize it the way you just did..

you learn more when you stray from abstraction, so clearly he's not trying to learn if he's using modules.

```
Code fromaddr = "example@gmail.com"
```

```
PASSWORD = "password"
```

here's an example of bad variable naming, a better suggestion would be

```
ADDR
```

```
or
```

```
EMAIL
```

and to shorten the PASSWORD variable, do

```
PASSWD
```

```
or
```

```
PASS
```

Code while True:

You could change this with

Code while x:

```
x -= 1
```

and have x be actually used.

then you can replace 'num' with 'x' and only use one variable, or you could completely remove 'x' and just use num and add loop handling.

Code (Press CTR + C to stop)

The fact it's asking to do this WITHOUT exception handling means the

Code server.quit()

will never get reached, therefore leaving an open connection.

going by your guidelines, let me suggest a better way of doing this, (without using while x:)

Code while True:

```
try:
```

```
<code>
```

```
except KeyboardInterrupt:
```

```
    server.quit()
```

```
    break
```

```
except Exception as exc:
```

```
    print(exc)
```

```
    break
```

Malicious Intent - May 13, 2017, 7:20 am

(May 13, 2017 07:14 AM)Chercher Wrote: [\[CITE\]](#) you learn more when you stray from abstraction, so clearly he's not trying to learn if he's using modules.

```
Code fromaddr = "example@gmail.com"
```

```
PASSWORD = "password"
```

here's an example of bad variable naming, a better suggestion would be

```
ADDR
```

```
or
```

```
EMAIL
```

and to shorten the PASSWORD variable, do

```
PASSWD
```

```
or
```

```
PASS
```

Code while True:

You could change this with

Code while x:

```
x -= 1
```

and have x be actually used.

then you can replace 'num' with 'x' and only use one variable, or you could completely remove 'x' and just use num and add loop handling.

Code (Press CTR + C to stop)

The fact it's asking to do this WITHOUT exception handling means the

```
Code server.quit()
```

will never get reached, therefore leaving an open connection.

going by your guidelines, let me suggest a better way of doing this, (without using while x:)

Code while True:

```
try:
```

```
<code>
```

```
except KeyboardInterrupt:
```

```
    server.quit()
```

```
    break
```

```
except Exception as exc:
```

```
    print(exc)
```

```
    break
```

I agree on error/exception handling and connection .. its definitely good practice to do so.

fromaddr is a variable from docs so cant really blame him for it ...

I would also replace the login details hardcoded with the prompt:

```
Code import getpass
```

```
user_name = input('Username:')
```

```
passw = getpass.getpass('Password:')
```

fiwhfivuehwu - June 4, 2017, 8:33 pm

Thanks for the share. This looks great.

