APT REPORTS INCIDENTS

The Equation giveaway

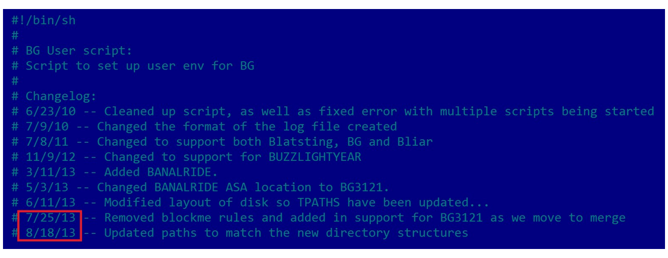
By GReAT on August 16, 2016. 7:22 pm

Rare implementation of RC5/RC6 in 'ShadowBrokers' dump connects them to Equation malware

August 13, 2016 saw the beginning of a truly bizarre episode. A new identity going under the name 'ShadowBrokers' came onto the scene claiming to possess files belonging to the apex predator of the APT world, the Equation Group [PDF]. In their initial leak, the ShadowBrokers claimed the archive was related to the Equation group, however, they didn't provide any technical details on the connections.

Along with some non-native rants against 'Wealthy Elites', the ShadowBrokers provided links to two PGP-encrypted archives. The first was provided for free as a presumptive show of good faith, the second remains encrypted at the time of writing. The passphrase is being 'auctioned', but having set the price at 1 million BTC (or 1/15th of the total amount of bitcoin in circulation), we consider this to be optimistic at best, if not ridiculous at face value.

The first archive contains close to 300MBs of firewall exploits, tools, and scripts under cryptonyms like BANANAUSURPER, BLATSTING, and BUZZDIRECTION. Most files are at least three years old, with change entries pointing to August 2013 the newest timestamp dating to October 2013.



```
#!/bin/sh
#
# BG User script:
# Script to set up user env for BG
#
# Changelog:
# 6/23/10 -- Cleaned up script, as well as fixed error with multiple scripts being started
# 7/9/10 -- Changed the format of the log file created
# 7/8/11 -- Changed to support both Blatsting, BG and Bliar
# 11/9/12 -- Changed to support for BUZZLIGHTYEAR
# 3/11/13 -- Added BANALRIDE.
# 5/3/13 -- Changed BANALRIDE ASA location to BG3121.
# 6/11/13 -- Modified layout of disk so TPATHS have been updated...
# 7/25/13 -- Removed blockme rules and added in support for BG3121 as we move to merge
# 8/18/13 -- Updated paths to match the new directory structures
```

As researchers continue to feast on the release, some have already begun to test the functional capabilities of the exploits with good results.

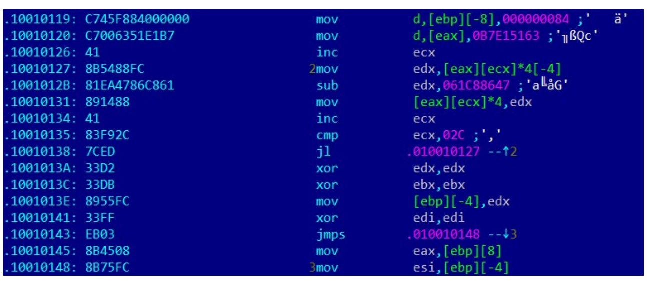
Having originally uncovered the Equation group in February 2015, we've taken a look at the newly released files to check for any connections with the known toolsets used by Equation, such as EQUATIONDRUG, DOUBLEFANTASY, GRAYFISH and FANNY.

While we cannot surmise the attacker's identity or motivation nor where or how this pilfered trove came to be, we can state that several hundred tools from the leak **share a strong connection** with our previous findings from the Equation group.

The Devil's in the Crypto

The Equation group uses the RC5 and RC6 encryption algorithms quite extensively throughout their creations. RC5 and RC6 are two encryption algorithms designed by Ronald Rivest in 1994 and 1998. They are very similar to each other, with RC6 introducing an additional multiplication in the cypher to make it more resistant. Both cyphers use the same key setup mechanism and the same magical constants named P and Q.

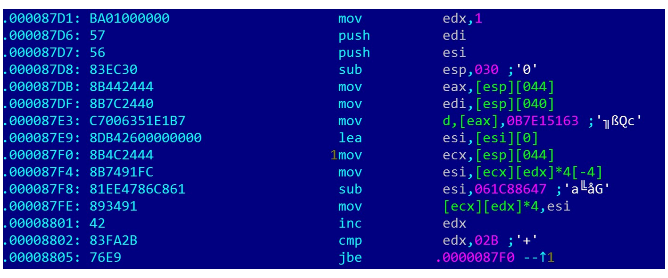
The particular RC5/6 implementation from Equation group's malware is interesting and deserves special attention because of its specifics. Inside the Equation group malware, the encryption library uses a subtract operation with the constant **0x61C88647**. In most publicly available RC5/6 code, this constant is usually stored as **0x9E3779B9**, which is basically **-0x61C88647**. Since an addition is faster on certain hardware than a subtraction, it makes sense to store the constant in its negative form and adding it instead of subtracting. In total, we've identified 20 different compiled versions of the RC5/6 code in the Equation group malware.



```
.10010119: C745F884000000          mov     d,[ebp][-8],000000084 ;'   ä'
.10010120: C7006351E1B7            mov     d,[eax],0B7E15163 ;'╖ßQc'
.10010126: 41                      inc     ecx
.10010127: 8B5488FC               2mov     edx,[eax][ecx]*4[-4]
.1001012B: 81EA4786C861            sub     edx,061C88647 ;'a╚åG'
.10010131: 891488                  mov     [eax][ecx]*4,edx
.10010134: 41                      inc     ecx
.10010135: 83F92C                  cmp     ecx,02C ;','
.10010138: 7CED                    jl      .010010127 --↑2
.1001013A: 33D2                    xor     edx,edx
.1001013C: 33DB                    xor     ebx,ebx
.1001013E: 8955FC                  mov     [ebp][-4],edx
.10010141: 33FF                    xor     edi,edi
.10010143: EB03                    jmps    .010010148 --↓3
.10010145: 8B4508                  mov     eax,[ebp][8]
.10010148: 8B75FC                 3mov     esi,[ebp][-4]
```

Encryption-related code in a DoubleFantasy (actxprxy32.dll) sample

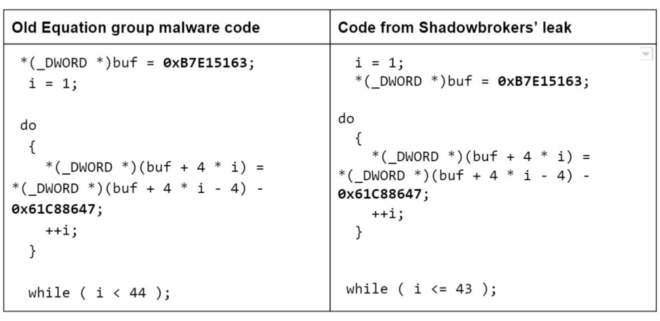
In the screenshot above, one can observe the main loop of a RC6 key setup subroutine extracted from one of the Equation group samples. The ShadowBrokers' free trove includes 347 different instances of RC5/RC6 implementations. As shown in the screenshot below, the implementation is functionally identical including the subtraction of the inverted constant **0x61C88647**.



```
.000087D1: BA01000000              mov     edx,1
.000087D6: 57                      push    edi
.000087D7: 56                      push    esi
.000087D8: 83EC30                  sub     esp,030 ;'0'
.000087DB: 8B442444                mov     eax,[esp][044]
.000087DF: 8B7C2440                mov     edi,[esp][040]
.000087E3: C7006351E1B7            mov     d,[eax],0B7E15163 ;'╖ßQc'
.000087E9: 8DB42600000000          lea     esi,[esi][0]
.000087F0: 8B4C2444               1mov     ecx,[esp][044]
.000087F4: 8B7491FC                mov     esi,[ecx][edx]*4[-4]
.000087F8: 81EE4786C861            sub     esi,061C88647 ;'a╚åG'
.000087FE: 893491                  mov     [ecx][edx]*4,esi
.00008801: 42                      inc     edx
.00008802: 83FA2B                  cmp     edx,02B ;'+'
.00008805: 76E9                    jbe     .0000087F0 --↑1
```

Specific RC6 implementation from "BUSURPER-2211-611.exe" (md5: 8f137a9100a9fcc8b512b3729878a373

Comparing the older, known Equation RC6 code and the code used in most of the binaries from the new leak we observe that they are functionally identical and share rare specific traits in their implementation.



Old Equation group malware code	Code from Shadowbrokers' leak

`*(_DWORD *)buf = 0xB7E15163;`
`i = 1;`
 	
`do`
`{`
`*(_DWORD *)(buf + 4 * i) = *(_DWORD *)(buf + 4 * i - 4) - 0x61C88647;`
`++i;`
`}`
 	
`while ( i < 44 );`
	`i = 1;`
	`*(_DWORD *)buf = 0xB7E15163;`
	
	`do`
	`{`
	`*(_DWORD *)(buf + 4 * i) = *(_DWORD *)(buf + 4 * i - 4) - 0x61C88647;`
	`++i;`
	`}`
	
	`while ( i <= 43 );`

In case you're wondering, this specific RC6 implementation has only been seen before with Equation group malware. There are more than 300 files in the Shadowbrokers' archive which implement this specific variation of RC6 in 24 different forms. The chances of all these being faked or engineered is highly unlikely.

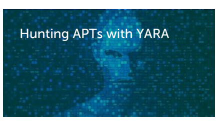
This code similarity makes us believe with a high degree of confidence that the **tools from the ShadowBrokers leak are related to the malware from the Equation group**. While the ShadowBrokers claimed the data was related to the Equation group, they did not provide any technical evidence of these claims. The highly specific crypto implementation above confirms these allegations.

More details about the ShadowBrokers leak and similarities with Equation group are available to Kaspersky Intelligence Services reports' subscribers. For more information, email intelreports@kaspersky.com

APT CYBER ESPIONAGE SHADOW BROKERS TARGETED ATTACKS VULNERABILITIES AND EXPLOITS ZERO-DAY VULNERABILITIES

Share post on:

Related Posts


Hunting APTs with YARA

Mokes and Buerak distributed under the guise of security certificates

Operation AppleJeus Sequel

THERE ARE 4 COMMENTS

Daniel
Posted on August 17, 2016. 9:44 pm

Remarkable piece

REPLY

Nielsen family
Posted on August 19, 2016. 10:03 am

Mega file not working

REPLY

John Smith
Posted on August 20, 2016. 1:11 pm

Although the dates in tar can be manipulated, are we confident it's not from Snowden's backups?

REPLY

Evgeny Yakovlev
Posted on August 23, 2016. 2:39 pm

You've probably already seen this, but anyway: https://www.cs.uic.edu/~s/musings/equation-group-rc6/

I've done some research and
GCC replaces add reg, imm with sub reg, -imm because -imm can be encoded in less space than imm. Here's a commit that introduced that for addsi_1 mnemonic: https://github.com/gcc-mirror/gcc/commit/c356ea4c37dbf32f7f40706f5eb2bf5d5129c77a

REPLY

LEAVE A REPLY

Your email address will not be published. Required fields are marked *

Enter your comment here

Name *

Email *

□ Save my name, email, and website in this browser for the next time I comment.

□ Notify me when new comments are added.

SUBMIT

□ I'm not a robot reCAPTCHA Privacy - Terms