Analiza zadania rekrutacyjnego

...

Analiza exploit kitu | RIG-v EK

Exploity:

- 3 exploity (2 na VBScript, Godmode)
- 1 exploit na flasha

Zadziałały 2 z nich

Persystencja | plik_co_dostalam.exe

1. Kopiuje się do c:\users\koko\appdata\roaming\44268089\svchost.exe
2. Zmienia uprawnienia powyższego pliku i katalogu na “tylko odczyt”
3. netsh advfirewall firewall add rule name="Quant" program=[PATH] dir=Out action=allow
4. Uruchamia plik

Persystencja | svchost.exe

Dodaje się do HKU\Software\Microsoft\Windows\CurrentVersion\Run Key:Quant
jeśli ten klucz nie istnieje, zamyka się jeśli istnieje

Komunikacja z C&C

1. Komunikacja z serwerem HTTP przez funkcję URLDownloadToFile(zapisane na stałe adresy (max 3))
   - rakuten24.ru/shop/index.php?id=44268089&c=1&mk=ca1a0f
   - rakuten24.ru/shop/index.php?id=44268089&c=2&mk=ca1a0f
   - itd ...

2. 3 możliwe komendy:
   - exe=url; -> WinExec("C:\\Users\\koko\\AppData\\Local\\Temp\\12655.exe")
   - doc=url; -> WinExec("C:\\Users\\koko\\AppData\\Local\\Temp\\12655.doc")
   - dll = crashuje

Crash komendy dll

kawałek funkcji która crashuje:

```
00402A08 | 83 EC 04                | sub esp,4
00402A0B | FF 85 F8 FD FF FF       | inc dword ptr ss:[ebp-208]
00402A11 | 89 44 24 04             | mov dword ptr ss:[esp+4],eax
00402A15 | 8B 45 08                | mov eax,dword ptr ss:[ebp+8]
00402A18 | 89 04 24                | mov dword ptr ss:[esp],eax
```

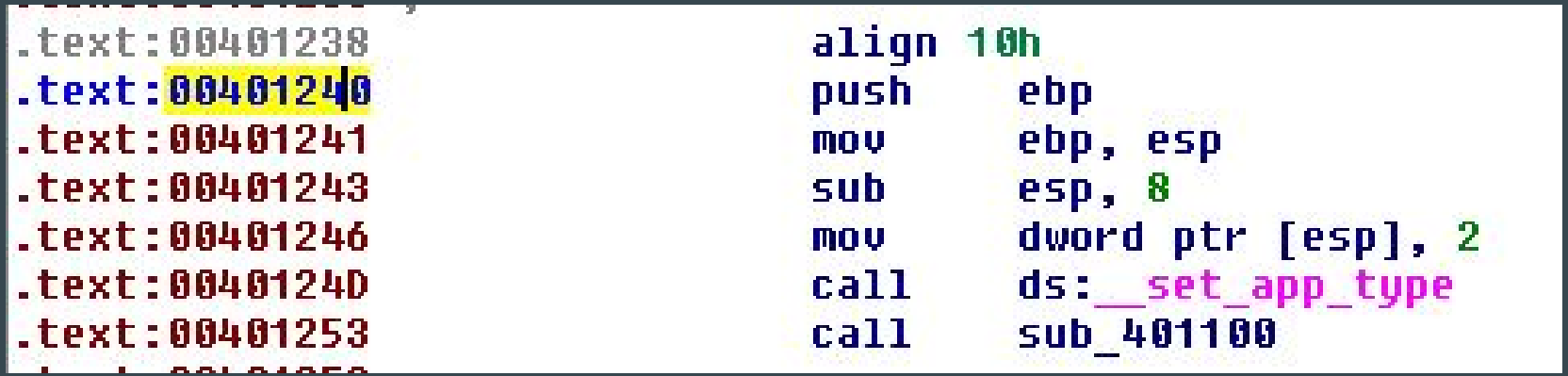
jak wyglądają podobne funkcje:

```
00402ADC | 83 EC 04                | sub esp,4
00402ADF | 8D 85 F8 FD FF FF       | lea eax,dword ptr ss:[ebp-208]
00402AE5 | 89 44 24 04             | mov dword ptr ss:[esp+4],eax
00402AE9 | 8B 45 08                | mov eax,dword ptr ss:[ebp+8]
00402AEC | 89 04 24                | mov dword ptr ss:[esp],eax
```

Utrudnianie analizy

- spakowane
- wewnętrzny mechanizm odszyfrowywania stringow

Utrudnianie analizy | spakowanie



Utrudnianie analizy | stringi

- Jedna funkcja która przyjmuje jako jedyny argument wskaźnik na zaszyfrowane
- Odkodowuje je zawsze tym samym kluczem zapisanym gdzieś w binarce

```
mov     [esp+18h+Dst], offset byte_404220 ; Str
call    decode_string   ; http://rakuten24.ru/shop/index.php
```

Utrudnianie analizy | stringi | bug w algorytmie szyfrowania

Algorytm deszyfrujący:

```
key_len = strlen(key);
for ( i = 0; strlen(encrypted_string) > i; ++i )
    decrypted_string[i] = encrypted_string[i] - key[i % key_len + 1];
```

Chyba jak powinien wyglądać:

```
key_len = strlen(key);
for ( i = 0; strlen(encrypted_string) > i; ++i )
    decrypted_string[i] = encrypted_string[i] - key[i % ( key_len + 1 )];
```

Utrudnianie analizy | stringi | skrypt w IDAPython

1. Odnalezienie funkcji deszyfrującej na podstawie sygnatury:

('push', 'mov', 'push', 'push', 'sub', 'mov', 'cmp', 'jnz', 'mov', 'jmp')

2. Odnalezienie klucza na podstawie referencji w funkcji deszyfrującej
3. Odszyfrowanie wszystkich stringów i wrzucenie ich w komentarze
4. Wyświetlenie na ekran adresów C&C i URLi

```
Found decoding function : 0x40145c
Found decoding key : 614b5e3c4536082dc6ec5f2a4029d69e
Found GetURLs function : 0x401cfc

Found C&C : http://kruibhez.ws/q/index.php
Found C&C : http://ufqeatci.org/q/index.php

Found url : http://kruibhez.ws/q/index.php?id=[machine_id]&c=[counter]&mk=554ddb
Found url : http://ufqeatci.org/q/index.php?id=[machine_id]&c=[counter]&mk=554ddb
```

reguła Yara

```
rule QuantRule
{
    meta:
        description = "Pliskal/Quant/Crugup"
        author = "Agnieszka Bielec"
        hash0 = "4a2fe144f831d12a0264170f09a2332f4a2624bfdcc6634972f45b5472ae50fa"
    strings:
        $s1 = "urlmon"
        $s2 = "URLDownloadToFileA"
        $s3 = "netsh advfirewall firewall add rule name=\""
        $s4 = "\" program=\""
        $s5 = "\" dir=Out action=allow"
        $s6 = "Quant"

    condition:
        all of them
}
```

Automatyczne wypakowywanie

- Zmodyfikowanie AppInit_DLLs
- DllMain instaluje hooka w __set_app_type na nową funkcję
- Ta funkcja wywołuje system("pd -pid %d -o quant_dumps")

Automatyczne wypakowywanie | hook | hot patching

msvcrt!__set_app_type

Automatyczne wypakowywanie | hook | kod

```
*(__set_app_type_addr+0) = 0xEB; //jmp $-5
*(__set_app_type_addr+1) = 0xF9; //jmp $-5

*(__set_app_type_addr-5) = 0xe9; //long jump. to the addres below
*((DWORD*)(__set_app_type_addr-4)) = (DWORD) ((unsigned char*)__set_app_type_new - __set_app_type_addr);
```