Understanding BumbleBee: The Malware Configuration and Clusters

vmray.com/cyber-security-blog/understanding-bumblebee-the-malware-configuration-and-clusters/

Understanding BumbleBee: BumbleBee’s malware configuration and clusters

Explore BumbleBee malware’s configuration secrets and discover the interconnected web of its malicious operations in this in-depth analysis.

BumbleBee Blog Series – 3

01 September 2023

DOWNLOAD THE E-BOOK

Table of Contents

Introduction

In our ongoing exploration of the enigmatic BumbleBee malware, we've previously dissected its delivery techniques, unraveled its malicious behavior, and delved into the ever-evolving nature of its evasion techniques.

Now, in this latest installment, we uncover the secrets hidden within the BumbleBee's malware configuration, shedding light on the methods it employs to safeguard its operations. Moreover, we'll take a comprehensive look into the clusters, where we'll connect the dots between different BumbleBee samples and missions.

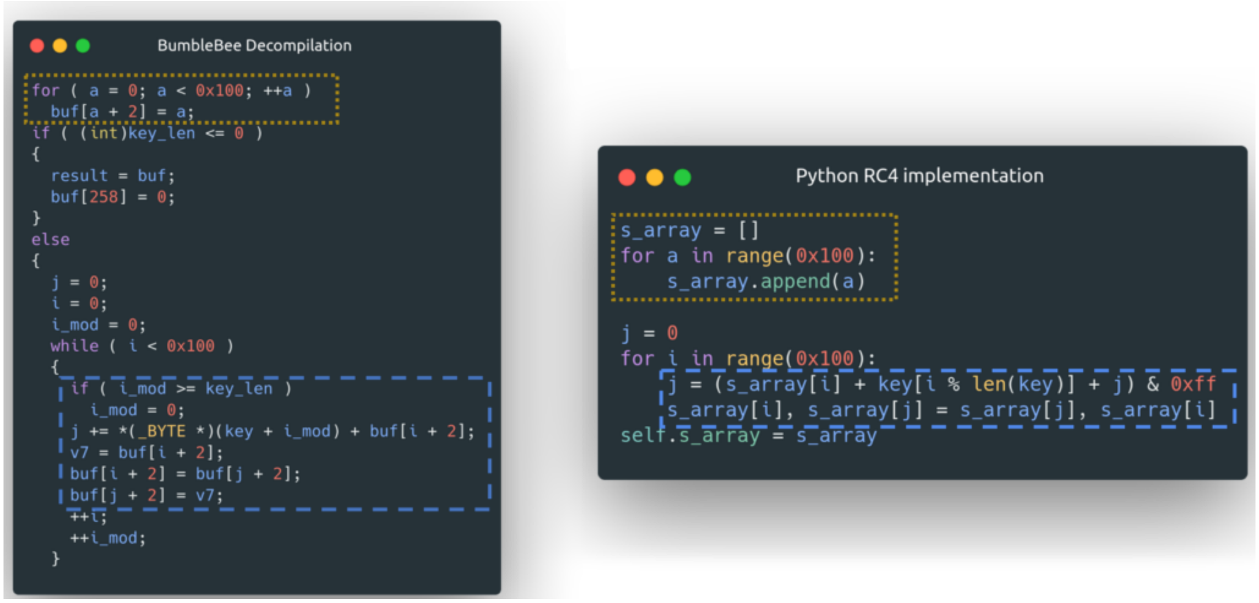
Join us as we explore BumbleBee's operations, revealing the hidden patterns that drive this malware's malicious activities.

Config Extractor

We have identified a number of differences between BumbleBee samples in terms of functionality but also how the config is processed. While some samples use no encryption at all, some use the RC4 algorithm to encrypt the configuration data.

To determine the encryption algorithm, we first had to locate the encrypted data as well as where it is further processed which allowed us to locate one function that looked promising. There are usually a few methods to identify which encryption algorithm a function implements. First, one can try to find magic constants strongly associated with certain encryption algorithms. For example, AES uses a so-called S-box filled with fixed constants to perform substitutions. These constants can easily be used to identify the algorithm. In BumbleBee's case, no such unique constants could be found. However, one commonly used encryption algorithm by malware, RC4, incidentally also contains no unique constants, making this a likely candidate.

In addition to the approach involving constants, another method is to compare the decompiled function to a list of known encryption algorithms, which in our case showed striking similarities to RC4 (see Figure 1).



BumbleBee Decompilation

```
for ( a = 0; a < 0x100; ++a )
  buf[a + 2] = a;
if ( (int)key_len <= 0 )
{
  result = buf;
  buf[258] = 0;
}
else
{
  j = 0;
  i = 0;
  i_mod = 0;
  while ( i < 0x100 )
  {
    if ( i_mod >= key_len )
      i_mod = 0;
    j += *(_BYTE *)(key + i_mod) + buf[i + 2];
    v7 = buf[i + 2];
    buf[i + 2] = buf[j + 2];
    buf[j + 2] = v7;
    ++i;
    ++i_mod;
  }
```

Python RC4 implementation

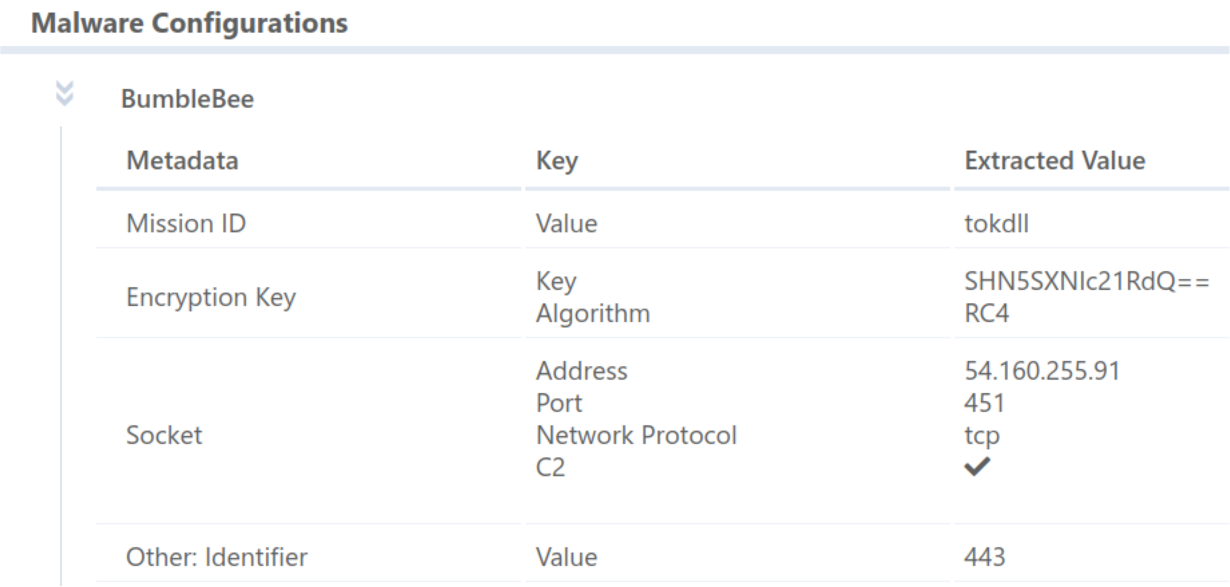
```
s_array = []
for a in range(0x100):
    s_array.append(a)

j = 0
for i in range(0x100):
    j = (s_array[i] + key[i % len(key)] + j) & 0xff
    s_array[i], s_array[j] = s_array[j], s_array[i]
self.s_array = s_array
```

Figure 1: Comparison between a decompiled function of BumbleBee showing similarities to RC4, thus allowing us to identify the encryption algorithm.

Finally, one can confirm the findings by decrypting the encrypted content via RC4 (using the extracted key) and investigating the output, e.g., via CyberChef or similar tools. While this confirmed our findings in this case, this method is of limited value if the malware authors decide to manipulate the encryption algorithm.

Investigating the decrypted configuration, we determined that BumbleBee generally has these configuration fields that we can export (see Figure 2):

- An RC4 key, if the config is encrypted
- A mission ID which can be freely set by the attacker, e.g., to distinguish different operations
- A list of C2 addresses, some of which are decoy addresses
- An “Identifier” often set to either 444 or 443, the main purpose of which is still undetermined – there is evidence that this is the port used to filter out the decoy addresses



Malware Configurations

BumbleBee

Metadata	Key	Extracted Value
Mission ID	Value	tokdll
Encryption Key	Key Algorithm	SHN5SXNlc21RdQ== RC4
Socket	Address Port Network Protocol C2	54.160.255.91 451 tcp ✓
Other: Identifier	Value	443

Figure 2: Configuration extracted from a BumbleBee sample.

Read the Ebook: Malware Configurations

Clustering

Based on this information, we tried to identify different samples and missions likely belonging to the same threat actors. For this purpose, we have randomly collected about a hundred BumbleBee samples that were seen for the first time in the wild from March to May 2023, extracted their configuration and plotted the relationship between them in clusters. While some of the samples seemed to be unique, i.e., they had a key and mission ID which were not shared with other samples, most samples had crossovers and shared mostly the same configuration.

During this analysis, we noticed that most samples belong to the same mission (“mc1905”), and even more use the same encryption key. Connected to this cluster are two other missions, “inst” and “mc1904” (see Figure 3). As all three missions share the same encryption key, we believe the same threat actor could be behind all of these samples.

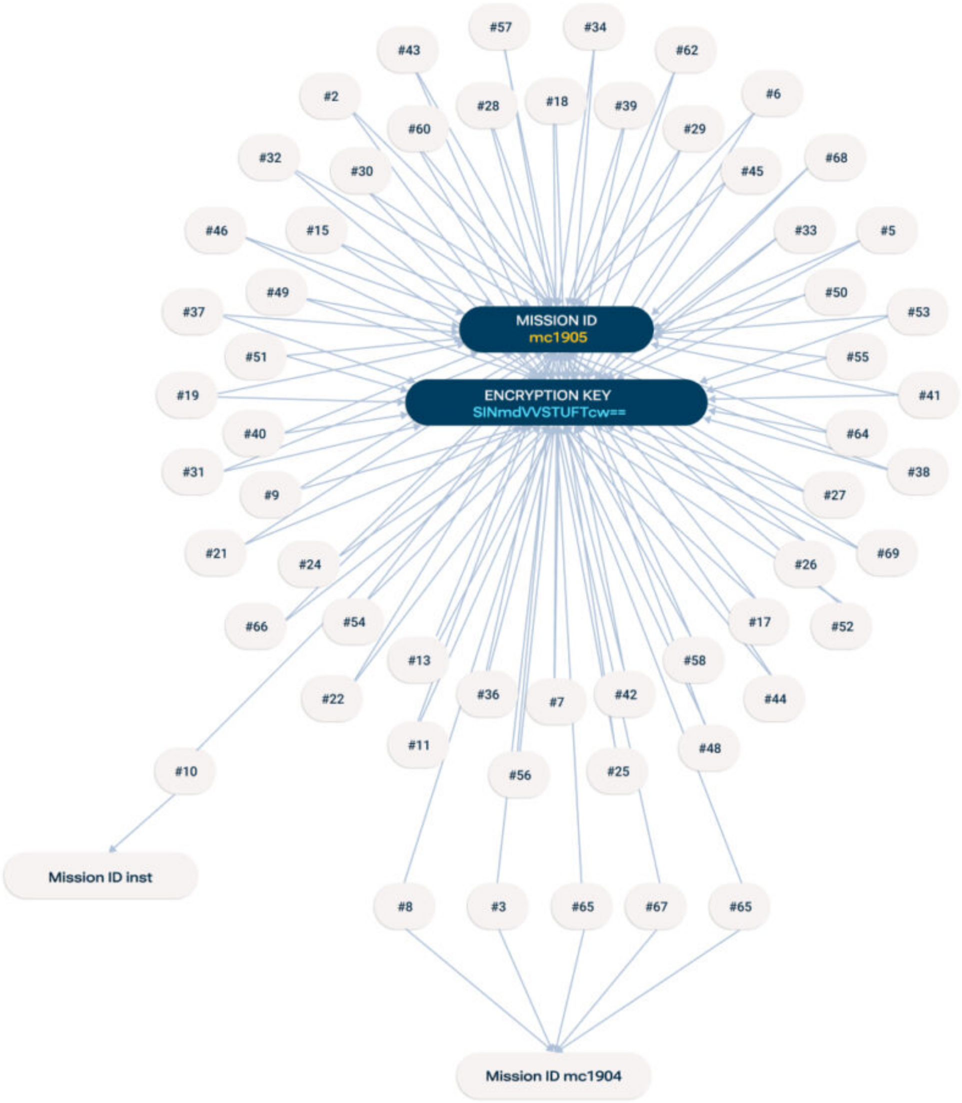


Figure 3: Biggest cluster found by analyzing about a hundred randomly selected BumbleBee samples seen in the last three months.

Furthermore, in Figure 4 we have plotted the samples that are unique or decoupled from the biggest cluster. Here, we have identified the missions “mvtm1703” and “0211r” to also likely stem from the same threat actor.

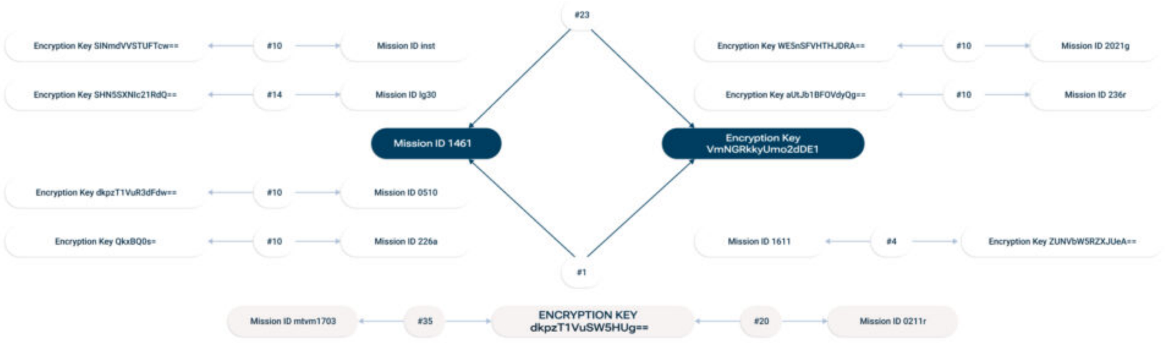

#23
Encryption Key SINmdVVSTUFTcw==
#10
Mission ID inst
Encryption Key SHN5SXNIc21RdQ==
#14
Mission ID lg30
Mission ID 1461
Encryption Key dkpzT1VuR3dFdw==
#10
Mission ID 0510
Encryption Key QkxBQ0s=
#10
Mission ID 226a
Encryption Key WESnSFVHTHJDRA==
#10
Mission ID 2021g
Encryption Key aUtJb1BFOVdyQg==
#10
Mission ID 236r
Encryption Key VmNGRkkyUmo2dDE1
Mission ID 1611
#4
Encryption Key ZUNVbW5RZXJUeA==
#1
Mission ID mtvm1703
#35
ENCRYPTION KEY dkpzT1VuSW5HUg==
#20
Mission ID 0211r

Figure 4: Plot of distinct missions found in the random subset of samples we have collected. Note the two samples on the bottom sharing the same encryption key while having different mission IDs.

This data reveals that in these three months, likely only one threat actor was dominating the space as most samples used the same encryption key (over 90% of the samples we looked at).

With API access to the VMRay Platform, the extraction of the config can be automatized to allow this kind of clustering and to follow threat actors over some period of time (see here).

Conclusion

Deep dives into malware families, as demonstrated in this blog post, help us to find better detection methods, proactively add VTI's to trigger on new malicious behavior and protect our customers before a dangerous malware becomes a threat.

This blog post also demonstrates how half a hundred evasion techniques are not enough to evade our dynamic, behavior-based analysis engine. Rather, we weaponize these efforts against the malware by trying to detect these attempts and revealing the malicious intent. As threat actors are always on the lookout for new methods to deliver their malware, regular updates of the VMRay Platform allow us to always be on track when it comes to new techniques.

References

https://research.checkpoint.com/2022/bumblebee-increasing-its-capacity-and-evolving-its-ttps/

https://www.cloudsek.com/blog/technical-analysis-of-bumblebee-malware-loader

https://elis531989.medium.com/the-chronicles-of-bumblebee-the-hook-the-bee-and-the-trickbot-connection-686379311056

IOCs

Hashes

June 2022

62a319d1b88070b6fc996226b2a213944f70f6e9370b89bcf761c6593420ae20

August 2022

5c15151a29fab8a2d58fa55aa6c88a58a456b0a6bc959b843e9ceb2295c61885

November 2022

2911bdd99140387cbc8761826aacc3c9de0ccb511255aa58790955d8337e2edf

December 2022

e81b21d6847961bc31a5446b556bde65234eb51cea23a2f928a2b79d13e35e03

January 2023

a41deed7a7bc99f4b45490e4572114b8cc2dd11f2301d954a59dee67fa3cca63

February 2023

897e53b648020ab28663240bbbce54546cf6f55b35019fd4aa2a209c4a3b1832

Analysis Report (February 2023)

51bb71bd446bd7fc03cc1234fcc3f489f10db44e312c9ce619b937fad6912656

ISO sample

c3148c6c4b0ecce9c7d07ba57dea96e35acf5f2ef47396c48339bb9a3a07e390

Emre Güler
Threat Researcher

BumbleBee Series – 1:
The delivery chains

BumbleBee Series – 2:
The malicious behavior