BlackWater Malware Leveraging Beirut Tragedy in New Targeted Campaign

quointelligence.eu/2020/08/blackwater-malware-leveraging-beirut-tragedy-in-new-targeted-campaign/

August 7, 2020

[Updated on August 11, 2020]

Executive Summary

- QuoIntelligence discovered a new targeted campaign delivering the novel BlackWater malware
- The campaign used the Beirut incident to lure the targets, with an attempt to spoof an official OSCE report
- The campaign was launched on 5 August, only one day after the Beirut incident. The provider hosting the attack infrastructure promptly shot down the C2 servers after our notification on 7 August
- The attack used state-of-the-art techniques, such as DNS over HTTPs (DoH)
- Based on the advanced TTPs and the discovered targets, we classify the threat actor as an Advanced Persistent Threat (APT) group named “ReconHellcat”
- Observed ReconHellcat targets consist of -at least- defense and diplomatic government bodies in Bulgaria and Azerbaijan

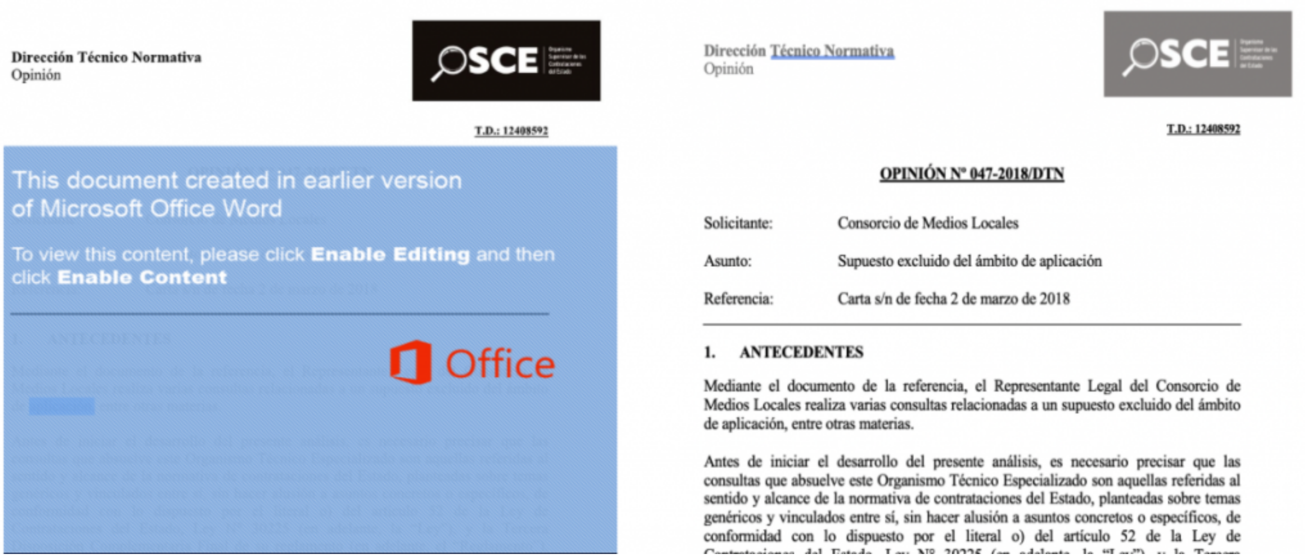
Introduction

On 6 August, QuoIntelligence discovered a malicious file uploaded to VirusTotal using the Beirut incident as a theme to lure the targets. Further analysis attributed the analysed artifact to BlackWater: a recently-discovered malware that uses Cloudflare Workers as a C2 communication interface and DNS over HTTPS (DoH) as name resolution channel.

We publicly disclosed our preliminary results on 7 August and made Cloudflare aware of the malicious infrastructure. Cloudflare promptly shutdown the C2 servers after our notification.

Technical Analysis

On 5 August, a compressed archive was uploaded to VirusTotal from Azerbaijan, containing a malicious, macro-enabled Word Document (maldoc) themed for the recent explosion in Beirut. The name of the maldoc (*047-20 – OSCE Report Beirut explosion.doc*) might indicate attackers’ intent to spoof the Organization for Security and Co-operation in Europe (OSCE), however, we discovered that attackers weaponized a legit Word document available in the web portal of the Organismo Supervisor de las Contrataciones del Estado (OSCE), which is a supervisory body of the Peruvian Ministry of Economy. The image below shows the comparison between the maldoc and the legit Word file from the Peruvian OCSE available in their website.



Dirección Técnico Normativa
Opinión

T.D.: 12408592

This document created in earlier version of Microsoft Office Word

To view this content, please click **Enable Editing** and then click **Enable Content**

Dirección Técnico Normativa
Opinión

T.D.: 12408592

OPINIÓN Nº 047-2018/DTN

Solicitante: Consorcio de Medios Locales

Asunto: Supuesto excluido del ámbito de aplicación

Referencia: Carta s/n de fecha 2 de marzo de 2018

1. ANTECEDENTES

Mediante el documento de la referencia, el Representante Legal del Consorcio de Medios Locales realiza varias consultas relacionadas a un supuesto excluido del ámbito de aplicación, entre otras materias.

Antes de iniciar el desarrollo del presente análisis, es necesario precisar que las consultas que absuelve este Organismo Técnico Especializado son aquellas referidas al sentido y alcance de la normativa de contrataciones del Estado, planteadas sobre temas genéricos y vinculados entre sí, sin hacer alusión a asuntos concretos o específicos, de conformidad con lo dispuesto por el literal o) del artículo 52 de la Ley de

The maldoc contains an obfuscated macro that retrieves the second-stage encrypted payload from one of the three hardcoded C2 URLs depending on the user's Windows version. To decrypt the payload, attackers implemented the old-fashion substitution cipher technique *Book cipher* using the legit Windows file `C:\Windows\Fonts\cambriab.tff` as key. To note that the content of this file differs depending on the Windows version , this is why the malicious macro checks the Windows version at the beginning of its execution, and contacts a specific URL depending of the result of this check. Finally, the decrypted payload is written in `%APPDATA%\\Microsoft\Word\STARTUP\Moon.wll` . By storing a Word add-in in such path allows Microsoft Word to load and execute it a malicious payload whenever it is launched. Although this technique (T1137/006) was firstly documented by F-Secure in 2017, only this year researchers publicly reported two APT groups firstly using it: Naikon APT and Vicious Panda.

The malicious `Moon.wll` contains another encrypted set of three different C2 URLs, and once executed by Word, it downloads an encrypted x64 executable as the final backdoor. The decryption technique used to decrypt the x64 backdoor still uses the Windows font file as key, however, the C2 URL is passed as an encrypted parameter which is decrypted via a different substitution cipher relying on an hardcoded alphabet as a key.

Finally, the Blackwater backdoor starts communicating with the Clouldflare Worker via JSON. Notably, both the loader and the backdoor resolve the C2 hostnames via DNS-over-HTTP (DoH) by using a built-in feature of libcurl. Only the malicious Macro is using the system's default DNS server to get the first-stage payload.

To note that all the samples we have analysed had a very low AV detection rate when they were firstly uploaded to VirusTotal.

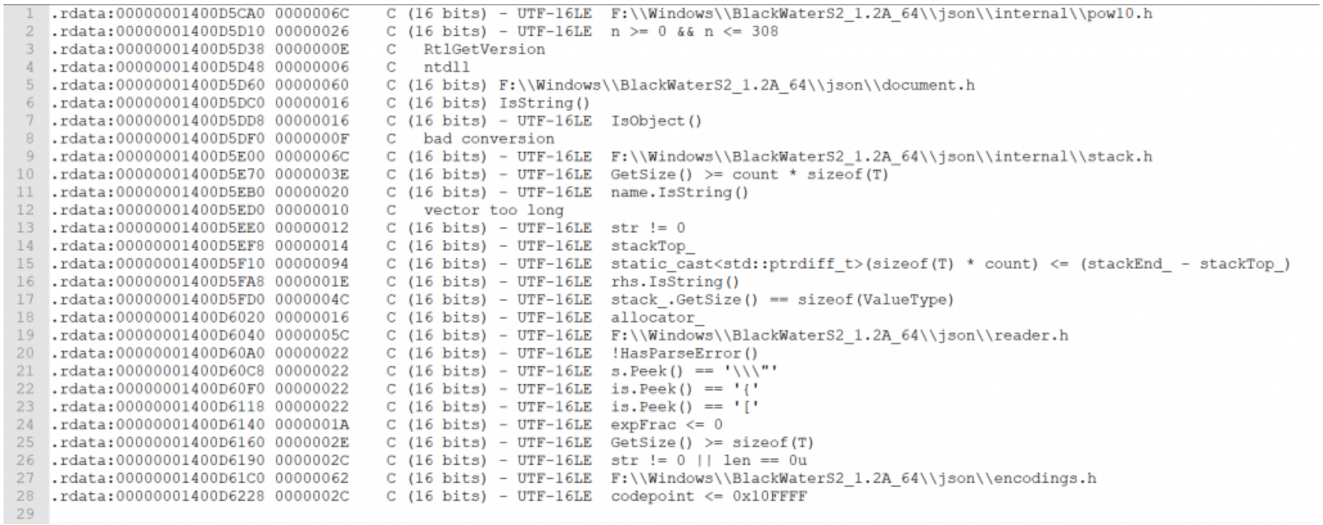
The C2 infrastructure was hosted on Cloudflare Workers platform, which is a service enabling users to register a subdomain and then have serverless code running to handle requests.

Attribution

Our technical findings overlap with the Tactics, Techniques and Procedures (TTPs) previously attributed to the BlackWater malware campaign first identified in March:

- Compressed archive containing a maldoc
- Blackwater strings within the malware
- DoH communications using `cloudflare-dns.com`
- C2 hosted on Cloudflare Workers service
- JSON encoded communications

As shown in the picture below, the analysis of the x64 sample revealed multiple strings containing the word “BlackWater”.



```
.rdata:00000001400D5CA0 0000006C    C (16 bits) - UTF-16LE  F:\\Windows\\BlackWaterS2_1.2A_64\\json\\internal\\pow10.h
.rdata:00000001400D5D10 00000026    C (16 bits) - UTF-16LE  n >= 0 && n <= 308
.rdata:00000001400D5D38 0000000E    C   RtlGetVersion
.rdata:00000001400D5D48 00000006    C   ntdll
.rdata:00000001400D5D60 00000060    C (16 bits) F:\\Windows\\BlackWaterS2_1.2A_64\\json\\document.h
.rdata:00000001400D5DC0 00000016    C (16 bits) IsString()
.rdata:00000001400D5DD8 00000016    C (16 bits) - UTF-16LE  IsObject()
.rdata:00000001400D5DF0 0000000F    C   bad conversion
.rdata:00000001400D5E00 0000006C    C (16 bits) - UTF-16LE  F:\\Windows\\BlackWaterS2_1.2A_64\\json\\internal\\stack.h
.rdata:00000001400D5E70 0000003E    C (16 bits) - UTF-16LE  GetSize() >= count * sizeof(T)
.rdata:00000001400D5EB0 00000020    C (16 bits) - UTF-16LE  name.IsString()
.rdata:00000001400D5ED0 00000010    C   vector too long
.rdata:00000001400D5EE0 00000012    C (16 bits) - UTF-16LE  str != 0
.rdata:00000001400D5EF8 00000014    C (16 bits) - UTF-16LE  stackTop_
.rdata:00000001400D5F10 00000094    C (16 bits) - UTF-16LE  static_cast<std::ptrdiff_t>(sizeof(T) * count) <= (stackEnd_ - stackTop_)
.rdata:00000001400D5FA8 0000001E    C (16 bits) - UTF-16LE  rhs.IsString()
.rdata:00000001400D5FD0 0000004C    C (16 bits) - UTF-16LE  stack_.GetSize() == sizeof(ValueType)
.rdata:00000001400D6020 00000016    C (16 bits) - UTF-16LE  allocator_
.rdata:00000001400D6040 0000005C    C (16 bits) - UTF-16LE  F:\\Windows\\BlackWaterS2_1.2A_64\\json\\reader.h
.rdata:00000001400D60A0 00000022    C (16 bits) - UTF-16LE  !HasParseError()
.rdata:00000001400D60C8 00000022    C (16 bits) - UTF-16LE  s.Peek() == '\\\"'
.rdata:00000001400D60F0 00000022    C (16 bits) - UTF-16LE  is.Peek() == '{'
.rdata:00000001400D6118 00000022    C (16 bits) - UTF-16LE  is.Peek() == '['
.rdata:00000001400D6140 0000001A    C (16 bits) - UTF-16LE  expFrac <= 0
.rdata:00000001400D6160 0000002E    C (16 bits) - UTF-16LE  GetSize() >= sizeof(T)
.rdata:00000001400D6190 0000002C    C (16 bits) - UTF-16LE  str != 0 || len == 0u
.rdata:00000001400D61C0 00000062    C (16 bits) - UTF-16LE  F:\\Windows\\BlackWaterS2_1.2A_64\\json\\encodings.h
.rdata:00000001400D6228 0000002C    C (16 bits) - UTF-16LE  codepoint <= 0x10FFFF
```

We researched the targets of both BlackWater campaigns (March and August) and concluded that -at least- defense and diplomatic government bodies in Bulgaria and Azerbaijan were highly likely targeted. Based on the advanced TTPs and the targeted institutions we classify the threat actor as an Advanced Persistent Threat (APT) group. Since we have not found yet any solid overlap with any already-known APT group, we are naming the attackers behind the two BlackWater campaigns as “ReconHellcat”.

Do you want to stay informed of cyber and geopolitical threats targeting *your* organization? Are you interested in receiving exclusive and unpublished intelligence?

Get in touch!

Appendix I

Early Indicators of Compromise

047-20 – OSCE Report Beirut explosion.rar
b4ed39b6852bce329686ec44e2e8e39b1c0be9b7095cec2e1682e6e2ef724c69
047-20 – OSCE Report Beirut explosion.doc
8b5ae22661e690b6689ea5894cb05e288c64927c2f0ccf8c11c814ad56968376

Second Stage – Office Plugin – moon1.wll

9b35a2bddc006ff5ab7262f5fb15f8c7ab9aee8f34bf02815f81f0c74222438b

Third-stage x64 (decrypted)

f984bc0896813553ec8b6dbb9b72769b3bbc8fbbe401566049df2e2061f1a829

https://b1[.]earth09[.]workers[.]dev/

https://b2[.]earth09[.]workers[.]dev/

https://b3[.]earth09[.]workers[.]dev/

https://moon3[.]earth09[.]workers[.]dev

https://moon1[.]earth09[.]workers[.]dev

https://moon2[.]earth09[.]workers[.]dev

Join Our Newsletter!

Subscribe to our newsletter to receive Weekly Intelligence Summaries, cyber news, and exciting updates!

Only valid business emails will be approved.