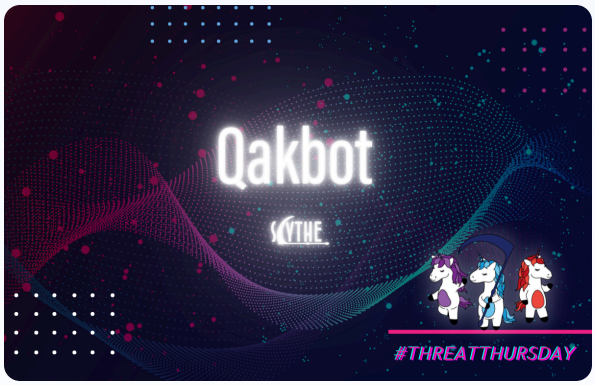
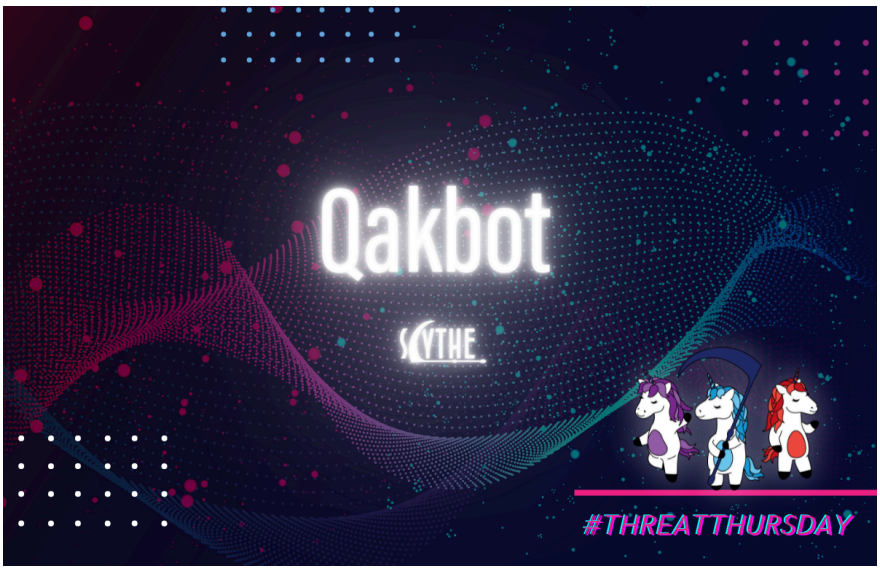
Threat Emulation: Qakbot



Intro Welcome to the July 2022 SCYTHE #ThreatThursday! This edition features an emulation of Qakbot, a piece of malware that is no stranger to the ...

Kristen Cotten

7 min. read • 28 Jul 2022



Latest Posts

Threat Thursday: September 2024

SEPTEMBER 30,2024

Threat Thursday:

Intro

Welcome to the July 2022 SCYTHE #ThreatThursday! This edition features an emulation of Qakbot, a piece of malware that is no stranger to the threat intel community. Qakbot, first seen in 2007, is making headlines again due to use by threat actors like Black Basta as a means of initial entry, lateral movement, and persistence.

Cyber Threat Intelligence

Profile:

Qakbot at its inception was a banking trojan that has been active since 2007. Over time it has evolved into a malware dropper that is commonly used by ransomware threat actors. As a result, detection of Qakbot can be considered a potential precursor to ransomware attacks due to its “malware installation-as-a-service” botnet. Qakbot’s modular nature makes it an appealing tool for threat actors as they can customize or build the payload according to the target of interest. This modularity makes defense a challenge as each Qakbot campaign can look slightly different on a given affected device.

Most recently (since at least July 11th), researchers report Qakbot is using the Windows 7 Calculator executable to infect devices using a technique known as DLL side-loading. Qakbot is spread via emails containing malicious links, documents, or embedded images. The most recent Qakbot emails contain an HTML attachment that downloads a password protected ZIP file. The password for the ZIP file is displayed on the HTML page. When the user opens the HTML email attachment, the ZIP download is triggered. The ISO file inside the ZIP is mounted when the user double clicks on it. Inside the ISO is a LNK file (link file) disguised as a PDF or other important document. This shortcut file launches the included calc.exe, which sideloads the Qakbot DLL.

Targets:

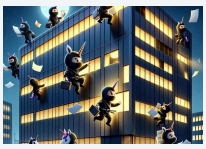
Although historically known as a banking trojan targeting the financial sector, Qakbot has evolved and expanded its reach. Due to its “malware

August 2024

AUGUST 31,2024

Threat Thursday: July 2024

JULY 31,2024



Threat Thursday: May 2024

MAY 30,2024

Threat Thursday: April

APRIL 25,2024

installation-as-a-service" Qakbot has been seen in multiple countries and across almost all continents-the Americas, Africa, Asia, and Europe.

Objectives:

Data theft, credential harvesting

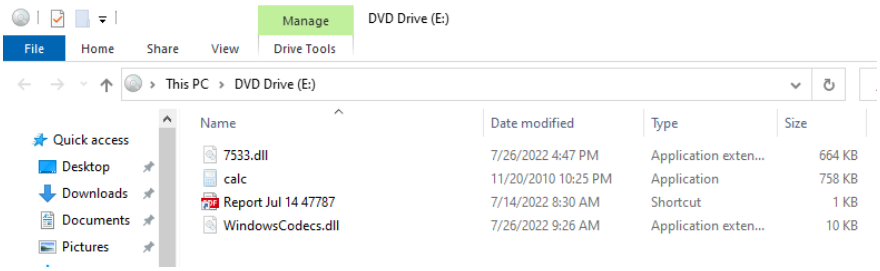
Capabilities:

- Reconnaissance
- Utilizes process injection to run a series of discovery commands:
- whoami /all, arp -a, ipconfig /all, net view /all, netstat -ano, net localgroup
- Lateral movement
- Windows Management Instrumentation (WMI)
- Privilege escalation and persistence
- Creation of scheduled tasks
- Registry manipulation
- Credential harvesting
- Attempts to enumerate credentials from multiple locations
- Targets browser data (including cookies and browser history)
- Data exfiltration
- Specifically exfiltrates emails
- Other payload delivery
- Cobalt Strike
- Often used by threat actors to deliver additional payloads or sell access to other threat actors

Attack

The automated emulation begins by downloading the ISO file from VFS. This ISO file was built by SCYTHE, but mimics one seen distributed in the wild via a password protected zip file embedded in an HTML email attachment. For the emulation, we've skipped the HTML email attachment and extracting the zip file since these offer little value in control validation.

The contents of the ISO file can be seen below. Note that all files except the LNK file have the hidden attribute set, so your display may differ.



The file calc.exe is a legitimate calculator executable, but it loads WindowsCodecs.dll. Because WindowsCodecs.dll is not listed in KnownDLLs, it can be sideloaded. The file WindowsCodecs.dll is a generic loader used by Qakbot, likely to add a step in the sideloading infection chain. The DLL was reverse engineered by SCYTHE analysts and a safe-for-distribution version was created.

The file 7533.dll is a Qakbot DLL that has been patched at a binary level to not execute any malicious instructions. The Qakbot loader invokes this DLL through regsvr32.exe, but oddly the DLL does not export the DllRegisterServer function, guaranteeing an error from regsvr32.exe. In practice, this error is suppressed because the DLL entry point never returns. Our patched version of the executable calls the Kernel32.dll export Sleep for two hours and then returns.

SCYTHE chose to patch the existing Qakbot DLL because it offers additional detection opportunities over just running a generic DLL named 7533.dll. After all, the code only differs by the few bytes that were patched. You can expect more of this in SCYTHE threats where we use patched malware to ensure things are safe for production while ensuring you get maximum detection opportunities.

To mimic the user clicking on the link file, SCYTHE developers created the file click-icon.exe (downloads as %USERPROFILE%\Desktop\not-for-detections.exe). This executable simulates clicking on the link with the appropriate working directory using the Windows ShellExecute API. You should not use this or the Mount-DiskImage PowerShell cmdlet for detection engineering purposes.

After executing regsvr32.exe and loading the malicious DLL (7533.dll), the SCYTHE implant executes a number of post-exploitation system recon commands, including:

- whoami /all
- cmd /c set
- arp -a
- ipconfig /all
- net view /all
- net share
- route print
- netstat -nao
- net localgroup

Each of these commands has been observed being executed by Qakbot threat actors. The Qakbot DLL distributed in this campaign is 32 bit, so ideally you'll want to run this campaign using a 32 bit executable. This has the practical implication of executing the post-exploitation commands from C:\windows\syswow64 rather than from C:\windows\system32.

Microsoft reported that Qakbot has recently used the C:\datop directory for staging of data, including the theft of emails. This directory is created in the emulation plan and fake email files are also populated in this directory. The 7-zip standalone command line binary is downloaded and used to archive the fake emails. This zip file is then exfiltrated to the SCYTHE server. Finally, the emulation removes the C:\Datop directory from the filesystem.

Clean Up Steps

Cleanup for this emulation requires the ISO file to be dismounted. This is done using the PowerShell cmdlet Dismount-Diskimage. The ISO file and SCYTHE-provided executable (not-for-detections.exe) are deleted.

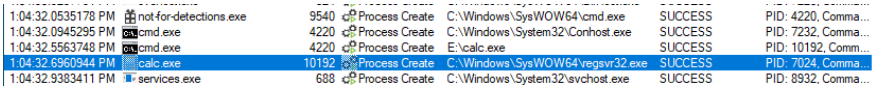
Finally, because the Qakbot DLL has been patched to sleep for two hours, the regsvr32 process will still be executing on most machines where the emulation is run. A PowerShell command is used to ensure that any regsvr32 process with 7533.DLL in the command line is terminated.

Detection Opportunities

Detection opportunities begin with the execution of calc.exe from a non-standard location. This is depicted in Procmon output below, but can also be seen with Sysmon event ID 1 and Security event ID 4688 (when configured).

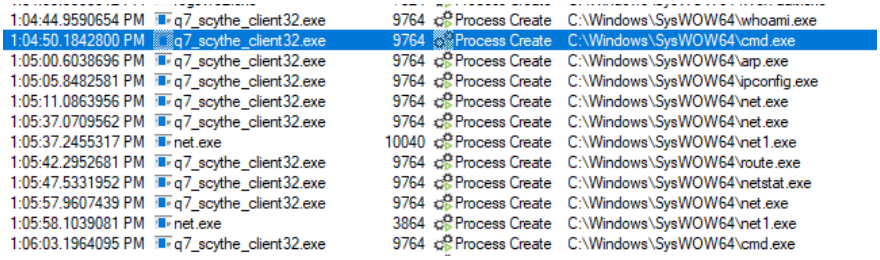
This calc.exe instance loads WindowsCodecs.dll, which should be expected to load from either C:\windows\system32 or C:\windows\syswow64 (depending on the architecture of the target binary, x86 in this case). In this case, it loads from the same folder as calc.exe itself (common in DLL sideloading attacks). Note in the graphic below that WindowsCodecs.dll is loaded late in execution for calc.exe (relative to legitimate DLLs). This can be detected through Sysmon event ID 7 (when configured).

The regsvr32.exe command is used to register components with the system. This should never be executed by calc.exe, regardless of where calc.exe is executing from. Process relationships can be examined via Sysmon event ID 1 and Security event ID 4688 (when configured).



1:04:32.0535178 PM	not-for-detections.exe	9540	Process Create	C:\Windows\SysWOW64\cmd.exe	SUCCESS	PID: 4220, Comma...
1:04:32.0945295 PM	cmd.exe	4220	Process Create	C:\Windows\System32\Conhost.exe	SUCCESS	PID: 7232, Comma...
1:04:32.5563748 PM	cmd.exe	4220	Process Create	E:\calc.exe	SUCCESS	PID: 10192, Comm...
1:04:32.6960944 PM	calc.exe	10192	Process Create	C:\Windows\SysWOW64\regsvr32.exe	SUCCESS	PID: 7024, Comma...
1:04:32.9383411 PM	services.exe	688	Process Create	C:\Windows\System32\svchost.exe	SUCCESS	PID: 8932, Comma...

Most system commands are not executed from C:\windows\syswow64 (as they will be if you execute this emulation using the x86 SCYTHE client). Alerting on commands such as “whoami.exe” or “ipconfig” running from C:\windows\syswow64 can provide early detection of x86 malware calling system commands, though this may have high false positive ratings. These can be detected through Sysmon event ID 1 and Security event ID 4688 (when configured).



1:04:44.9590654 PM	q7_scythe_client32.exe	9764	Process Create	C:\Windows\SysWOW64\whoami.exe
1:04:50.1842800 PM	q7_scythe_client32.exe	9764	Process Create	C:\Windows\SysWOW64\cmd.exe
1:05:00.6038696 PM	q7_scythe_client32.exe	9764	Process Create	C:\Windows\SysWOW64\arp.exe
1:05:05.8482581 PM	q7_scythe_client32.exe	9764	Process Create	C:\Windows\SysWOW64\ipconfig.exe
1:05:11.0863956 PM	q7_scythe_client32.exe	9764	Process Create	C:\Windows\SysWOW64\net.exe
1:05:37.0709562 PM	q7_scythe_client32.exe	9764	Process Create	C:\Windows\SysWOW64\net.exe
1:05:37.2455317 PM	net.exe	10040	Process Create	C:\Windows\SysWOW64\net1.exe
1:05:42.2952681 PM	q7_scythe_client32.exe	9764	Process Create	C:\Windows\SysWOW64\route.exe
1:05:47.5331952 PM	q7_scythe_client32.exe	9764	Process Create	C:\Windows\SysWOW64\netstat.exe
1:05:57.9607439 PM	q7_scythe_client32.exe	9764	Process Create	C:\Windows\SysWOW64\net.exe
1:05:58.1039081 PM	net.exe	3864	Process Create	C:\Windows\SysWOW64\net1.exe
1:06:03.1964095 PM	q7_scythe_client32.exe	9764	Process Create	C:\Windows\SysWOW64\cmd.exe

Sigma Detection Opportunities

Below are the Sigma rules applicable to the procedures carried out in the emulation plan. If any of the procedures covered in this section do not trigger an alert in the environment, it is recommended to deploy the relevant rule. Note, as with many Sigma rules, this set of rules may need to be baselined for your unique environment and filters added for approved activity by certain users, systems, or applications.

Suspicious Calculator Execution

The first coverage provided by Sigma is to detect suspicious activity around the calculator application. To detect this, the first rule, Suspicious Process Parents, looks for uncommon processes being spawned by calc.exe. The next detection opportunity in this execution phase is the Suspicious Calculator Usage rule which looks for calc.exe coming from suspicious directories such as a mounted ISO.

Whoami

The first action on objective procedure mapped to Sigma is in step eleven of the emulation plan.

In this step, the whoami execution maps to the following three Sigma rules:

- Local Accounts Discovery
- Whoami Execution
- Whoami Execution Anomaly

Suspicious Network Commands

The next detection opportunities are presented in steps thirteen, fourteen, and seventeen conducting network enumeration as pictured below.

All of these network enumeration steps map to the Suspicious Network Command Sigma Rule.

Net Command

Continuing enumeration procedures takes us to our next set of detection opportunities around net command usage and its related applications, net.exe and net1.exe. It is in steps fifteen, sixteen, and nineteen of the emulation, the net command is used.

Here we find that the Net.exe Execution Sigma rule applies to the three procedures due to their command parameters. Step fifteen will also trigger a second rule, Windows Network Enumeration.

Netstat

The final Sigma detection opportunity is the usage of netstat in step eighteen.

For this procedure, the applicable Sigma rule is Suspicious Listing of Network Connections.

Respond

If any of the alerts are detected in the environment, the response team should determine the depth of the Kill Chain, collect artifacts, and answer the following questions:

- Was the installation successful?
- What are the persistent mechanisms?
- Is Command & Control (C2) successful?
- What are the domain names, IP addresses, ports, and protocols used?
- Are there observations of Actions on Objectives (AOO)?
- What are they?
- Did the actor laterally move?
- Was sensitive data taken?
- Usernames, Passwords, Other?
- What caused the initial compromise?
- How was it delivered?
- What was exploited?
- Vulnerability, Control, Human?

Once it has been determined how deep the intrusion goes, containment, eradication, and recovery should begin. After recovery, lessons learned should drive additional courses of action (COAs) to thwart the threat should it return, such as implementing additional security controls. As always, please follow your organization's response plan and evidence retention policies. We also recommend leveraging NIST SP 800-61 Rev. 2.

This Threat Thursday post discusses active research by SCYTHE and other cited third parties into an ongoing threat. The information in this post should be considered preliminary and may be updated as research

continues. This information is provided "as-is" without any warranty or condition of any kind, either express or implied.

About the Authors

Jake Williams and Kristen Cotten of SCYTHE's CTI team contributed to this report and the creation of the threat emulation. Christopher Peacock assisted with QA and performed detection engineering.

About SCYTHE

SCYTHE provides an advanced attack emulation platform for the enterprise and cybersecurity consulting market. The SCYTHE platform enables Red, Blue, and Purple teams to build and emulate real-world adversarial campaigns in a matter of minutes. Customers are in turn enabled to validate the risk posture and exposure of their business and employees and the performance of enterprise security teams and existing security solutions. Based in Arlington, VA, the company is privately held and is funded by Gula Tech Adventures, Paladin Capital, Evolution Equity, and private industry investors.

References

- New Variant of QakBot Being Spread by HTML File Attached to Phishing Emails (fortinet.com)
- Technical analysis of the QakBot banking Trojan | Securelist
- A closer look at Qakbot's latest building blocks (and how to knock them down) - Microsoft Security Blog
- QAKBOT Loader Returns With New Techniques and Tools (trendmicro.com)

Related Threat Thursday

Threat Thursday - NetWire RAT

Threat Thursday - NetWire RAT Happy Fall everyone! I'm Christopher Peacock, the newest unicorn to...

Read More ›

Threat Actor APT35

Executive Summary Iranian threat actor(s) have been observed using PowerShell modules and unmanaged...

Read More ›

Qakbot Reloaded

Qakbot Reloaded

Read More ›