QakBot Banking Trojan Causes Massive Active Directory Lockouts

Light Dark

June 2, 2017

By Mike Oppenheim, Kevin Zuk, Matan Meir, Limor Kessem

10 min read

Advanced Threats Banking & Finance Fraud Protection Incident Response Threat Intelligence

Research for this blog was facilitated by X-Force IRIS contributors Mark banal.

lockouts caused hundreds to thousands of AD users to get locked out of their company's domain in rapid succession, leaving employees of the impacted organizations unable to access their endpoints, company servers and networked assets.

Active Directory manages users and user access on Microsoft servers, as well as the policies and procedures that enable network access. X-Force researchers associated the mass AD lockouts with malicious activity by an existing banking Trojan known as QakBot, aka PinkSlip.

X-Force Incident Response and Intelligence Services (IRIS) responders, who investigated recent QakBot activity waves, suspect that numerous organizations have suffered and will continue to suffer from these lockout waves.

QakBot Back in Business

According to X-Force research, QakBot is financial malware known to target businesses to drain their online banking accounts. The malware features worm capabilities to self-replicate through shared drives and removable media. It uses powerful information-stealing features to spy on users' banking activity and eventually defraud them of large sums of money.

Though well-known and familiar from previous online fraud attacks, QakBot continually evolves. This is the first time IBM X-Force has seen the malware cause AD lockouts in affected organizational networks.

Although part of QakBot is known to be a worm, it is a banking Trojan in every other sense. QakBot is modular, multithread malware whose various components implement online banking credential theft, a backdoor feature, SOCKS proxy, extensive anti-research capabilities and the ability to subvert antivirus (AV) tools. Aside from its evasion techniques, given admin privileges, QakBot's current variant can disable security software running on the endpoint.

Overall, QakBot's detection circumvention mechanisms are less common than those used by other malware of its class. Upon infecting a new endpoint, the malware uses rapid mutation to keep AV systems guessing. It makes minor changes to the malware file to modify it and, in other cases, tire code to make it appear unrecognizable.

QakBot's Dropper Run

Much like other malware of its class, the QakBot Trojan is ushered into infected endpoints through a dropper. The dropper typically uses delayed execution to evade detection. It lands on the target endpoint and halts before any further action for 10 to 15 minutes, hoping to elude sandboxes that might try to analyze it upon arrival. Next, the dropper opens an explorer.exe instance and injects the QakBot Dynamic Link Libraries (DLL) into that process.

After deployment, the dropper corrupts its original file. It uses the ping.exe utility to invoke a ping command that will repeat six times in a loop:

C:\Windows\System32\cmd.exe" /c ping.exe -n 6 127.0.0.1 & type "C:\Windows\System32\autoconv.exe" à "C:\Users\UserName\Desktop\7a172.exe

When the pings are complete, the contents of the original QakBot dropper are overwritten by the legitimate Windows autoconv.exe command. (Autoconv.exe converts file allocation table (FAT) and FAT32 volumes to the NTFS file system, leaving existing files and directories intact at startup after Autochk runs.) A snippet of QakBot's JavaScript downloader is shown below:

Figure 1: QakBot downloader script

In the example above, the download locations that would fetch the QakBot payload were lightly obfuscated using character codes. The downloader from this sample attempted to connect to the following three update servers:

- projects[.]montgomerytech[.]com
- n[.]abcwd0.seed.fastsecureservers[.]com
- css.kbaf.myzen[.]co[.]uk

At the time we ran the sample, the following download server responded with a large amount of ASCII hex, which was the QakBot payload: projects[.]montgomerytech[.]com/TealeafTarget.php.

Figure 2: QakBot obfuscated payload

autoconv.exe utility.

To communicate with infected machines via rendezvous domains created on the fly, QakBot uses both a Domain Generation Algorithm (DGA) and a list of hardcoded command-and-control (C&C) servers. The hardcoded C&C servers observed in our sample are included at bottom of this article.

Persistence Mechanisms

QakBot is notorious for its capability to persist on infected machines. This, combined with the malware's AD lockout capabilities, makes it especially frustrating to detect and remove in enterprise environments.

To keep itself alive after system reboots and removal attempts, QakBot establishes persistence mechanisms on the target systems using a Registry runkey and scheduled tasks. It creates a "\CurrentVersion\Run" registry entry to automatically launch itself after each new run of the operating system. An example run key created by the malware was "HKCU\Software\Microsoft\Windows\CurrentVersion\Run\xyhz," which pointed to "C:\Users\UserName\AppData\Roaming\Microsoft\Graroaojr\graroaoj.exe."

QakBot adds another layer of persistence and creates recurring, named, scheduled tasks via "schtasks.exe" to run itself on timed intervals and ensure that it has not been disabled or removed.

Figure 3: QakBot Process tree showing schtasks.exe and ping usage.

QakBot typically creates two named scheduled tasks. The first scheduled task periodically launches QakBot:

- C:\Windows\system32\schtasks.exe" /create /tn {1F289CDD-BD80-4732-825C-4D2D43DA75AB} /tr "\"C:\Users\UserNameUserName\AppData\Roaming\Microsoft\Graroaojr\graroaoj.exe\"" /sc HOURLY /mo 7 /F

The second scheduled task launches a separate JavaScript based downloader with a .wpl extension:

```
C:\Windows\system32\cscript.exe //E:javascript
\"C:\Users\UserName\AppData\Local\Microsoft\graroaoj.wpl\"\"" /sc
WEEKLY /D TUE /ST 12:00:00 /F
```

Resident QakBot's Active Directory Lockouts

While observing QakBot's effect on the Active Directory domain, X-Force researchers witnessed the malware performing the following three activities:

1. Locking out hundreds to thousands of accounts in quick succession;
2. Automated logon attempts, which can be launched using accounts that do not exist (i.e., argo, operator, administrador, user, prof, owner, usuario, admin, HP_Administrator, HP_Owner, Compaq_Owner, Compaq_Administrator, etc.); and
3. Deploying malicious executables to network shares and registering them as a service.

Guessing User Credentials Until Lockout

To spread through the affected network, QakBot can move laterally, both automatically and on demand, by a remote command from its C&C server. To activate that capability, the attacker launches the malware's command "13," also known as "nbscan" in earlier variants of QakBot.

To access and infect other machines in the network, the malware uses the credentials of the affected user and a combination of the same user's login and domain credentials, if they can be obtained from the domain controller (DC). QakBot may collect the username of the infected machine and use it to attempt to log in to other machines in the domain. If the malware fails to enumerate usernames from the domain controller and the target machine, the malware will use a list of hardcoded usernames instead.

Figure 4: QakBot's hardcoded usernames.

To authenticate itself to the network, the malware will attempt to match usernames with various passwords. We have observed three password schemes, which may serve to defeat weak or default passwords:

- The password is the reversed username (for example, username = administrator; password = rotartsinimda).
- The username is tested with various hardcoded passwords in a dictionary attack style.

Figure 5: QakBot's hardcoded password strings used in dictionary attack style.

Below is the assembly of a stack frame to attempt a connection to the IPC$ administrative share of a target machine using a username obtained from a domain controller:

Figure 6: Attempted connection to the IPC$ administrative share.

IPC$ is part of the common hidden network shares that are accessible only to administrators. Attackers may use it in conjunction with administrator-level credentials to remotely access a networked system over server message block (SMB). Usually, the purpose is to interact with systems using remote procedure calls, transfer files and run transferred binaries through remote execution, which could help QakBot run its malicious code.

Once the malware successfully connects to the IPC$ administrative share of the target machine, it checks to determine whether it can create a service locally. If it can, QakBot proceeds to enumerate the network shares of the target machine and then attempts to drop a copy of itself to one of the shares. Once a copy of the malware is dropped, the malware creates and starts a service in the target machine to execute it.

Under certain domain configurations, the malware's dictionary attack for accessing the target machines can result in multiple failed authentication attempts, which eventually trigger an account lockout.

Figure 7: Accounts lockouts logged.

Enter Banking Trojan Mode

work.

QakBot implements man-in-the-browser (MitB) functionality that allows injected malicious code to be inserted into online banking sessions. Instead of keeping them inside its configuration file, QakBot fetches these malicious scripts on the fly from the domain it controls, in the following format:

- hxxps://[AttackerDomain/wbj/br/content/TargetBankName/TargetBankName.js

These scripts are commonly referred to as webinjections because they are used to manipulate the visual content that infected users see on their banking websites. The code snippet below, labeled “WIRE” by the author, appears to check whether “To enroll in the” is visible on the wire transfer page of the targeted bank.

This is very typical Trojan behavior, designed to figure out where to start inserting the malicious code to modify the page and match the fraud M.O. the attacker has planned. It’s easy to see in this example that QakBot is targeting corporate banking services and aiming to reach the “change address” page of the compromised account.

Figure 8: QakBot webinjections targeting corporate banking accounts.

Another snippet from the same webinjection script seeks to collect personal information displayed in the online banking session by querying the document object model (DOM) elements of the page with names that are known to house sensitive details, such as date of birth and Social Security number.

Figure 9: QakBot webinjections harvest victim personally identifiable information (PII).

Information Stealing Modules

The malware’s operators typically use QakBot to piggyback on banking sessions initiated by the user. QakBot’s theft mechanisms allow it to steal information including:

- ...es;

- HTTP(S) session authentication data;
- Cookies, including authentication tokens and Flash cookies; and
- FTP and POP3 credentials.

Other data typically exfiltrated by QakBot and sent to a criminal-controlled FTP server include:

- System information;
- IP address;
- Domain Name System (DNS) name;
- Host name;
- Username;
- Domain;
- User privilege;
- OS version;
- Network interfaces (address, netmask and status);
- Installed software;
- Credentials from the endpoint's protected storage;
- Account name and webserver credentials;
- Connection type;
- POP3 username, server and password; and
- SMTP server and email addresses.

Typical Online Propagation

QakBot propagation in the wild most often takes place via exploit kits (EKs) and spam campaigns that target employees rather than widespread webmail users. Once inside the network, QakBot acts as a worm that can spread through network shares and removable drives.

In terms of magnitude, researchers reported that a recent QakBot botnet had successfully militarized over 54,000 infected computers.

QakBot's Targets

Discovered in the wild in 2009, QakBot is historically considered one of the most advanced banking Trojans active in the wild. It is also the first Trojan that was designed to exclusively target the business banking sector, a [illegible] it has kept true throughout the past eight years.

banking. X-Force IRIS responders have seen QakBot attacks in the pharmaceutical and technology sectors.

Figure 10: Current QakBot configuration by target type (Source: IBM X-Force).

According to X-Force researchers, QakBot's operators have been upgrading the malware's code, persistence mechanisms, anti-AV and anti-research capabilities. As the malware evolves, it has also been known to target organizations in the health care and education sectors.

Researchers believe that a closed, organized cybercrime gang with roots in Eastern Europe is responsible for QakBot.

Global Perspective

From a global perspective, QakBot's focus on the business sector and its periods of inactivity leave it at the bottom of the top 10 list of the most active malware families. In the past five years, the group operating QakBot has been in and out of the cybercrime arena, likely in an attempt to keep attacks to a minimum and avoid law enforcement attention.

Figure 11: Top most prevalent financial malware families (Source: IBM X-Force, May 2017 YTD).

Mitigating QakBot Infections

To detect threats such as QakBot, banks and service providers should use adaptive malware detection solutions that provide real-time insight into fraudster techniques and address the relentless evolution of the threat landscape.

Keeping QakBot out of employee endpoints starts with cybersecurity awareness, since this malware may come through infected websites or via email attachments. Users can protect themselves and their organizations by practicing browsing hygiene, disabling online ads, filtering macro execution in files that come via email and observing other security best practices.

update frequently used programs and delete those no longer in use. To mitigate QakBot activity on the network, make sure domain accounts are configured with the least privilege required to perform job tasks.

Organizations can also create a random domain admin account for safety purposes and ensure that it reports directly to the security information and event management (SIEM) system upon any attempt to use it. A special emergency account can enable security staff to recover service and determine the source when network users are being locked out.

Finally, prevent workstation-to-workstation communications where possible to force malware out of the trenches and into areas where central detection systems will pick it up quickly.

Indicators of Compromise

Malware dropper MD5:

- 2C5901F06E6211BB7F6D22AB3376C22C
- A1593E2DC521EA8F66BB727B4725EC2C

Malware sample MD5:

- 8a3ab5d3fa3644ec1829e7825b0a22a3
- 08BACFFCC1E4DF896670047790373497
- 847BCDB4F5C2EBA96E8943430C4402C8

QakBot's hardcoded C2 servers:

- 96[.]67[.]244[.]225:443
- 96[.]3[.]92[.]39:443
- 173[.]31[.]254[.]105:443
- 192[.]158[.]217[.]32:993
- 181[.]165[.]242[.]18:443
- 90[.]211[.]106[.]62:443
- 70[.]97[.]146[.]196:443
- 50[.]124[.]113[.]135:443
- 97[.]64[.]195[.]106:993
- 47[.]21[.]79[.]34:443
- 47[.]21[.]79[.]34:465
- 05[.]52:2222

- 89[.]43[.]179[.]209:443
- 180[.]93[.]148[.]41:443
- 68[.]115[.]254[.]146:443
- 73[.]166[.]43[.]103:443
- 209[.]136[.]9[.]64:443
- 76[.]17[.]137[.]223:443
- 47[.]21[.]79[.]34:995
- 74[.]65[.]227[.]38:443
- 24[.]91[.]39[.]131:2222
- 74[.]101[.]41[.]97:443
- 50[.]134[.]209[.]66:443
- 24[.]45[.]150[.]163:443
- 24[.]123[.]151[.]58:443
- 76[.]8[.]200[.]134:443
- 105[.]227[.]251[.]148:443
- 132[.]206[.]59[.]132:443
- 211[.]27[.]18[.]233:995
- 174[.]51[.]185[.]121:465
- 24[.]184[.]200[.]177:2222
- 96[.]67[.]244[.]225:443
- 184[.]90[.]203[.]138:995
- 68[.]53[.]54[.]125:443
- 98[.]113[.]137[.]220:443
- 86[.]27[.]41[.]234:443
- 91[.]93[.]4[.]222:443

- 50[.]101[.]245[.]7:2222

Banking Malware | Banking Security | Banking Trojan | Financial Industry | Financial Malware | Malware | Online Banking | Trojan | X-Force