Google Drive
Remote Access Trojan

SNIFFING OUT RATS IN THE CLOUD

Malware Report | February 2016
# TABLE OF CONTENTS

Google Drive Remote Access Trojan (RAT) ................................................................. 1

Key Findings .................................................................................................................. 1

Technical Analysis of Attack Flow .............................................................................. 3

Google API .................................................................................................................. 3

Malware Analysis ........................................................................................................ 4

GDRAT .......................................................................................................................... 4

IMapping GDRAT to the IXESHE Campaign ............................................................... 5

Bottom Line .................................................................................................................. 10

Actionable Threat Intelligence .................................................................................... 11

Related MD5 Hashes .................................................................................................... 12

Yara Rules .................................................................................................................... 12

Verint, Powering Actionable Intelligence® ................................................................. 13

Unauthorized use, duplication, or modification of this document in whole or in part without the written consent of Verint Systems Inc. is strictly prohibited.

By providing this document, Verint Systems Inc. is not making any representations regarding the correctness or completeness of its contents and reserves the right to alter this document at any time without notice.

Features listed in this document are subject to change. Please contact Verint for current product features and specifications.

All marks referenced herein with the ® or TM symbol are registered trademarks or trademarks of Verint Systems Inc. or its subsidiaries. All rights reserved. All other marks are trademarks of their respective owners.

© 2016 Verint Systems Inc. All rights reserved worldwide.
As cloud computing grows in popularity, many organizations are using a variety of cloud services, such as Google Drive, to improve collaboration and information sharing. However, this added convenience doesn’t come without a price – namely, the increased exposure to tech-savvy cyber attackers.

Malware that uses Google Drive to communicate with attackers and hide malicious files is not new and has been the subject of various security vendor reports over the past few years. However, a fresh twist to this malware was recently discovered by the Verint research team during an investigation of a government organization breach.

For the first time on record, this investigation unearthed undisputable evidence that connects the dots between the Google Drive malware and the nefarious IXESHE campaign, which has been targeting government organizations for several years. An in-depth analysis of the tactics, techniques and procedures (TTPs) used by the malware, described below, revealed the use of encrypted command & control channels and other stealthy communication techniques to avoid blacklisting by network monitoring tools.

In addition to exposing the Chinese attacker group, the Verint research team also uncovered threat intelligence in the form of Gmail accounts that could be used by perimeter tools to thwart future attacks. In addition, our findings enabled us to warn two additional unsuspecting government organizations that they were being breached.

Since perimeter security products, such as network gateways and anti-viruses, are unable to analyze encrypted content and hardly ever block “trusted” Google services, the Google Drive Remote Access Trojan (GDRAT) operated in the background for years without (in the vast majority of instances) ever triggering an alert from anti-virus solutions.

We discovered that GDRAT had been active within the target organization since 2013. Over the past two years, it had uploaded 3471 files, with a grand total of 881 MB of information. Not only that, while investigating this malware, we also discovered in Google Drive exfiltrated files from two other government organizations targeted by the IXESHE attacker group.

Surprisingly, our analysis of GDRAT log files revealed some login records from well-known cyber security research organizations and sandbox services. This led us to conclude that these firms did not analyze malware in a quarantined environment, which may have exposed the host information of their analysis personnel.
GOOGLE DRIVE REMOTE ACCESS TROJAN (RAT)

Key Findings
Several key findings during the investigation enabled to achieve an in-depth understanding of the malware's TTPs, which in turn produced clear evidence of the link between GDRAT and the IXESHE cyber campaign.

Detailed Login Records
GDRAT maintained a daily log of login records, using names, login time, and time zone of compromised hosts. Since these records were meticulously organized according to dates, we were able to learn about both the scope and time of the attacks. We also discovered that almost all recent samples, from 2014 onward, used the UPX shelling technique to avoid detection by static engines.

Breach Duration
Malware samples of GDRAT found on the host computers we investigated dated back to as early as 2011. This finding testifies to the stealthy and persistent nature of this APT.

Use of Gmail Accounts
Our analysis showed that one Gmail account would be used on no more than three victim organizations. This fact led us to discover two additional organizations (besides the organization that commissioned the investigation) that, unbeknownst to them, were also breached.

Suspicious Registered Domains
Among the C2 communications we discovered, PassiveDNS data showed that one IP (202.4.112.235) corresponded with three domains (as shown in the table below). The registered name alice yoker and its e-mail chuni_fan@sina.com were mentioned in an Arbor Network research in association with registered domains that were often used in IXESHE malware.

<table>
<thead>
<tr>
<th>Domain Registered</th>
<th>Name</th>
<th>E-mail</th>
</tr>
</thead>
<tbody>
<tr>
<td>apple365.asseyas.com</td>
<td>SITELUTIONS REGISTRAR</td>
<td><a href="mailto:DOMAINS@SITELUTIONS.COM">DOMAINS@SITELUTIONS.COM</a></td>
</tr>
<tr>
<td>volume.yahoobigdeals.com</td>
<td>alice yoker</td>
<td><a href="mailto:Qinyz001@163.com">Qinyz001@163.com</a></td>
</tr>
<tr>
<td>showrecord.actionews.net</td>
<td>alice yoker</td>
<td><a href="mailto:chuni_fan@sina.com">chuni_fan@sina.com</a></td>
</tr>
</tbody>
</table>

Table 1 - PassiveDNS data
TECHNICAL ANALYSIS OF THE ATTACK FLOW

Reconnaissance

Our investigation also confirmed that Active Directory (AD) accounts and passwords with high-level access credentials were leaked. Since AD is a tool used by system administrators to manage end user Windows computers, it is reasonable to assume that these credentials belonged to system administrators. By configuring AD Group Policy, attackers were able to deploy Logon Script - a feature used by system administrators to define tasks to be performed when a user logs on to a computer - to launch their malware on the compromised hosts. We also have reason to believe that these attackers conducted comprehensive reconnaissance prior to waging their attack. They evidently had access to personnel information within the organization, as they were able to precisely target the computers of senior management personnel and their secretaries. Moreover, the attackers were also aware of the applications (such as Chrome, Skype, and PDFCreator) most commonly used by organization personnel, and deployed malware with file names similar to such applications, placing them in normal directories.

Google API

Google provides an API that allows users to develop applications for Google cloud services. Users must first register an application in the Google Developer Console, before obtaining a token that authorizes access to the Google services.

To obtain an access token, users have to send an HTTP POST to the Google API. The HTTP POST must include four parameters: client_id, client_secret, refresh_token, and grant_type2 (as shown in Figure 1).

![Figure 1 - The Access Token for Google API](image)
Malware Analysis

GDRAT

GDRAT used the Google Drive API to upload data from the compromised computers to the attackers' registered Google Drive storage. To avoid causing an anomaly in the hosts' network traffic (and thus draw attention to a possible security breach) by uploading a massive number of files most of which are worthless to attackers, GDRAT only targeted files (Word, Excel, PowerPoint, and PDF) that had been opened or edited by the users (as shown in Figure 2). It did this by monitoring the Recent folder (as shown in Figure 3). When the file status in the folders changes, GDRAT uploads the relevant file to Google Drive. Files named ~DF7690.tmp or ~FD7K25.tmp would then appear in the %TEMP% folder (as shown in Figure 4).

![Figure 2: GDRAT only synchronized certain file formats](image)

<table>
<thead>
<tr>
<th>Operating Systems</th>
<th>Path</th>
</tr>
</thead>
<tbody>
<tr>
<td>Windows XP</td>
<td>%PROFILE%\Recent</td>
</tr>
<tr>
<td>Windows 7</td>
<td>%AppData%\Microsoft\Windows\Recent</td>
</tr>
</tbody>
</table>

![Figure 3: Location of “Recent” in different operating systems](image)

<table>
<thead>
<tr>
<th>ASCII</th>
<th>Path</th>
</tr>
</thead>
<tbody>
<tr>
<td>00FC16A8</td>
<td>20006320 &quot;extList is doc,xls,ppt,pdf,docx,pptx,xlsx&quot;,LF</td>
</tr>
<tr>
<td>00FC16AC</td>
<td>0000002A</td>
</tr>
<tr>
<td>00FC16E0</td>
<td>00000044</td>
</tr>
<tr>
<td>00FC16E4</td>
<td>00FC171C</td>
</tr>
<tr>
<td>00FC16E8</td>
<td>00000001</td>
</tr>
</tbody>
</table>

![Figure 5: Google Drive API OAuth parameters](image)

2 Using OAuth 2.0 to Access Google APIs
https://developers.google.com/identity/protocols/OAuth2
With the four parameters we acquired from GDRAT, we used a restful API to obtain the access token from Google, which gave us access to the attackers’ Google Drive storage data (as shown in Figure 7). In one of the malware samples, we discovered that GDRAT had been exfiltrating information out of the organization since 2013. Over the past two years, it had uploaded 3471 files, with a grand total of 881 MB of information (as shown in Figure 8). Another sample showed that attackers would organize the uploaded files daily, moving old files into an “OLD” folder and placing them in chronological order (as shown in Figure 9).
Figure 8 - GDRAT had been infiltrating the organization since 2013

Figure 9 - Attackers organize the uploaded files daily
Every time one of the compromised hosts logged into Google Drive, GDRAT would establish a daily log of its login records, using names, login time, and time zone of the compromised hosts as file names (as shown in Figure 10). In one GDRAT sample, we discovered a few login records from cyber security research organizations and sandbox services. These researchers and service providers did not analyze malware in a quarantined environment, which not only made the host information of analysis personnel vulnerable to leaks, but also allowed attackers to check for login anomalies through login records. We also used the login records to analyze the scope and time of the attacks.

Figure 10- Login records from cyber security research organizations and sandbox services
Because GDRAT was deployed through the Google API, which encrypts all communications using HTTPS by default, it was very difficult for various network gateways to detect the malicious traffic. Since these network-level security products are unable to analyze encrypted content and hardly ever block "trusted" Google services, in most cases these attacks slipped by without triggering an alert from anti-virus solutions. Out of the 15 GDRAT malware samples we found, the most effective anti-virus software only detected a little more than half (as shown in Figure 11) of these samples.

<table>
<thead>
<tr>
<th>File</th>
<th>Ratio</th>
<th>First sub.</th>
<th>Last sub.</th>
</tr>
</thead>
<tbody>
<tr>
<td>432bfb0d1337628ba03005b98a7950202d99409e1e51cc6b30e058bc320965e93829d1</td>
<td>6 / 56</td>
<td>02.47.56</td>
<td>02.47.56</td>
</tr>
<tr>
<td>a462aeb3b70a24e4b961f6d71b19eed0aa23e835d9c2d521eb17cc89d44113160549760d9e020526952473c388</td>
<td>18 / 53</td>
<td>03.28.54</td>
<td>07.40.06</td>
</tr>
<tr>
<td>55f207d99616e6c1d752057281973ac3ff1f1a7543210ad33ce2f4bc2f384bd0153175adb001656499b0c9f45</td>
<td>9 / 57</td>
<td>03.29.06</td>
<td>03.29.06</td>
</tr>
<tr>
<td>d9e57059f9c2f8482f096a8e2b5b2b6f8b062f65531571250cf5dcb58ab89fb42433130a682c9363a4a61fa3184574</td>
<td>33 / 57</td>
<td>14:58:59</td>
<td>06:35:11</td>
</tr>
<tr>
<td>9ae5b6a3f6b8e820777c8779665e2b05b120f65531571250cf5dcb58ab89fb42433130a682c9363a4a61fa3184574</td>
<td>26 / 55</td>
<td>07:31:36</td>
<td>10:17:16</td>
</tr>
<tr>
<td>e932ca99220d0e0805d95c3e97c1424a88ce595fe1a54df0ace6e9c8cb573a630e86385399331c2940b638c104f</td>
<td>3 / 42</td>
<td>10:39:10</td>
<td>10:39:10</td>
</tr>
<tr>
<td>89f58e1a8a06b1a0218e8153e8ec4e334fcd80cc58011833cbb52433e488042b08e58b8ef9d75bb62a663371d8450c3</td>
<td>9 / 56</td>
<td>03:31:14</td>
<td>03:31:14</td>
</tr>
<tr>
<td>26040d6be9b2e1e1b4c755bc08dd7c9b9db2ab38417de29198d0428b06666475f231a7d14884e9b975a5b52201df5</td>
<td>9 / 57</td>
<td>02.19.59</td>
<td>02.30.01</td>
</tr>
</tbody>
</table>

*Figure 11- Antivirus software detection rates*
Mapping GDRAT to the IXESHE Campaign

The Verint research team investigation revealed detailed and comprehensive evidence related to the workings of the GDRAT malware. By understanding its TTPs, we were able to link it to the IXESHE campaign.

In previous cases involving the IXESHE malware, the hosts’ system information would be encoded and encrypted in the malware (for example Base64, RC4, and RSA), and then sent to a hard-coded command & control server via HTTP connections. Therefore, by analyzing network traffic, we could identify its pattern and find the compromised hosts. In this investigation, however, we discovered several innovations in this particular strain of malware:

• **Command & control information was no longer hard-coded within the malware.** In previous attacks, attackers hard-coded command & control information in the malware. A new tactic used in this attack was to obtain command & control information through external parameters and carry out remote scheduling through AT(?) commands (as shown in Figure 12). If the command & control was detected and blocked, attackers could reset the schedule to assign a new command & control.

• **The malware was embedded with an SSL certificate.** Using String Stacking, a technique commonly used in IXESHE malware, the certificate was assembled in the memory (as shown in Figure 13) and certified by the attackers themselves (as shown in Figure 14). IXESHE used this certificate to communicate with its command & control server, establishing an encrypted SSL connection to avoid cyber security detections. We refer to this as the “IXESHE Tunnel”.

Because command & control information was not hard-coded in the IXESHE Tunnel, blacklisting could not effectively detect the malware. Furthermore, the malware communicated with command & control via an embedded certificate and SSL encrypted connection, which made it impossible to find its pattern through traditional network traffic analysis.

Nevertheless, our research team was able to recognize the IXESHE Tunnel as an attack of the highest severity level (as shown in Figure 15). We were able to detect compromised hosts within the organization and help the organization remediate the attack in the quickest possible manner.

![Figure 12 - Attackers scheduling through AT commands](image-url)
Google Drive Remote Access Trojan

Figure 13 - IXESHE Tunnel SSL Certificate

Figure 14 - Content of Certificate

Figure 15 - IXESHE Tunnel Endpoint Forensics Detection Results
Bottom Line
Given the sophistication of today’s APTs, signature-based anti-virus and network protection products are no longer reliable or effective enough in detecting advanced cyber threats within organizations. Constant real-time monitoring and analysis of payloads, network traffic and endpoints, together with on-demand forensics and automated investigations, are required for better protection of critical information assets.
Actionable Threat Intelligence

We recommend that organizations update their perimeter tools with the following threat intelligence to protect against GDRAT:

### Related MD5 Hashes

<table>
<thead>
<tr>
<th>No</th>
<th>MD5</th>
<th>Attacker Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>EFD7628F9365E4ACBE855600BB2412DBD</td>
<td><a href="mailto:wmingimg@gmail.com">wmingimg@gmail.com</a></td>
</tr>
<tr>
<td>2</td>
<td>47F5231A7D14884E9B975A5B52201DF5</td>
<td><a href="mailto:rater.huang@gmail.com">rater.huang@gmail.com</a></td>
</tr>
<tr>
<td>3</td>
<td>93B1D589265A81B5F68CD6D96AB8EAC5</td>
<td><a href="mailto:rater.huang@gmail.com">rater.huang@gmail.com</a></td>
</tr>
<tr>
<td>4</td>
<td>EFE24777391948DCD44806FE44FFB90C</td>
<td><a href="mailto:weichuan260@gmail.com">weichuan260@gmail.com</a></td>
</tr>
<tr>
<td>5</td>
<td>38A292F587F4F0E3F4FF7E47CC1C9F21</td>
<td><a href="mailto:alarng.wu@gmail.com">alarng.wu@gmail.com</a></td>
</tr>
<tr>
<td>6</td>
<td>07B13745DA54861D231ECD7825F4015</td>
<td><a href="mailto:atmovies.wang@gmail.com">atmovies.wang@gmail.com</a></td>
</tr>
<tr>
<td>7</td>
<td>566FE51F8E6E57DFD25F7B975B9B1D0C</td>
<td><a href="mailto:kely1120@gmail.com">kely1120@gmail.com</a></td>
</tr>
<tr>
<td>8</td>
<td>58BBE8FFD975BB62BA2E63371D8450C3</td>
<td><a href="mailto:maroon5.2033@gmail.com">maroon5.2033@gmail.com</a></td>
</tr>
<tr>
<td>9</td>
<td>63A9E863FC5389931C42940B638C1E4F</td>
<td><a href="mailto:blocktest2010@gmail.com">blocktest2010@gmail.com</a></td>
</tr>
<tr>
<td>10</td>
<td>B42433130A682C936E34A61FA3184574</td>
<td><a href="mailto:cn.toyang@gmail.com">cn.toyang@gmail.com</a></td>
</tr>
<tr>
<td>11</td>
<td>3B2F60E243DA49D347FA12B8F053D668</td>
<td><a href="mailto:shardan.chen@gmail.com">shardan.chen@gmail.com</a></td>
</tr>
<tr>
<td>12</td>
<td>CF02B9A07958F8CD1C01293C0FA536AB</td>
<td><a href="mailto:shardan.chen@gmail.com">shardan.chen@gmail.com</a></td>
</tr>
<tr>
<td>13</td>
<td>31E05E4D9769DE2E0825629524473C88</td>
<td><a href="mailto:himikoran@gmail.com">himikoran@gmail.com</a></td>
</tr>
<tr>
<td>14</td>
<td>BDO153175ADA800165B395D61C9FFF45</td>
<td><a href="mailto:himikoran@gmail.com">himikoran@gmail.com</a></td>
</tr>
<tr>
<td>14</td>
<td>1CAA95F775DA673DB5EC08459B86C252</td>
<td>Account Disable</td>
</tr>
</tbody>
</table>
Yara Rules

rule GDRAT_Recent
{
  strings:
  $menu1 = "--05 Version--"
  $menu2 = "==Recent Dir=="
  condition:
    any of them
}

rule GDRAT_Error
{
  strings:
  $err1 = "##### ChainFilter PANIC ######"
  $err2 = "##### EveryService PANIC ######"
  $err3 = "##### MsgHandler PANIC ######"
  $err4 = "##### BackgroundService PANIC ######"
  $err5 = "##### Service PANIC ######"
  condition:
    any of them
rule GDRAT_Drop
{
  strings:
  $drop1 = "-OF690b.tmp"
  $drop2 = "-FD6255.tmp"
  condition:
    any of them
}

rule GDRAT_UPX
{
  strings:
  $hex1 = (28 00 84 BE 00 F0 FF BB 00 10 00 00 00 FF 6A 04 53 57 FF D5
          8D 07 9F 01 00 00 00 00 00 00 00 6A 04 53 57 FF)
  $hex2 = (28 00 05 BA 07 9F 01 00 00 00 00 00 00 00 6A 04 53 57 FF 68)
  $hex3 = (28 00 84 74 07 8D 08 83 C3 04 EE 81 FF 06 78)
  condition:
    all of them
}
About Verint Systems Inc.
Verint® (Nasdaq: VRNT) is a global leader in Actionable Intelligence® solutions with a focus on customer engagement optimization, security intelligence, and fraud, risk and compliance. Today, more than 10,000 organizations in 180 countries — including over 80 percent of the Fortune 100 — count on intelligence from Verint solutions to make more informed, effective and timely decisions.

www.verint.com/cyber | info.cyber@verint.com