CNACOM - Open Source Exploitation via Strategic Web Compromise

Published on:
December 01, 2016

Authored by:

Ed Miles

Category:

APT

CNACOM - Open Source Exploitation via Strategic Web Compromise

Introduction

Since a full proof of concept for CVE-2016-0189 vulnerability was published on GitHub, Zscaler ThreatLabZ has been closely tracking its proliferation. The first copying of the exploit code we spotted was from the Sundown exploit kit (EK), followed closely by Magnitude and a resurgent KaiXin EK. In addition to the commoditized EKs, this exploit code has been leveraged in numerous one-shot and gated web-exploitation campaigns, delivered through a mix of the usual malvertising networks and compromised websites.

This blog details CNACOM, a web-based campaign that appears to be related to a well-known nation-state actor more commonly associated with spear-phishing attacks.

Infection Cycle

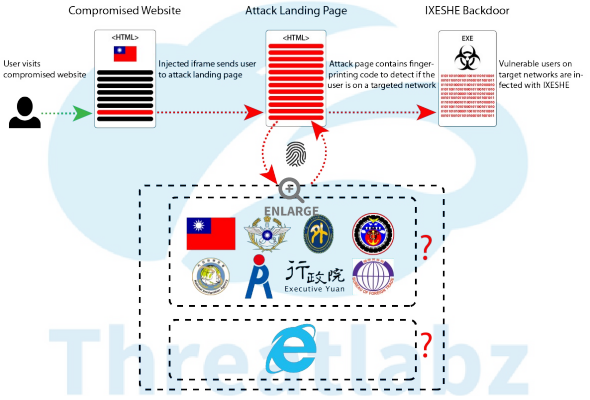

Figure 1 - An overview of the campaign's infection flow, highlighting the targeted organizations

On November 7, we spotted a malicious injection on the registration page of a major Taiwanese public service website. An iframe was injected into the footer of the page, which then loaded a unique landing page containing the CVE-2016-0189 exploit code.

```
635     <map name="Map">
636         <area shape="RECT" coords="297,66,421,82" href="
637             alt="">
638     </map>
639   <iframe name="abc" width=0 height=0      frameborder=0 
640 </body>
641 </html>
```

Figure 2 - An injected iframe with the name "abc" redirects visitors to the attack code

The landing page, hosted on an RHCloud virtual private server (VPS), begins with a nearly identical copy of the GitHub-published code, though the payload invocation appears to use a sandbox escape via CVE-2015-0116.

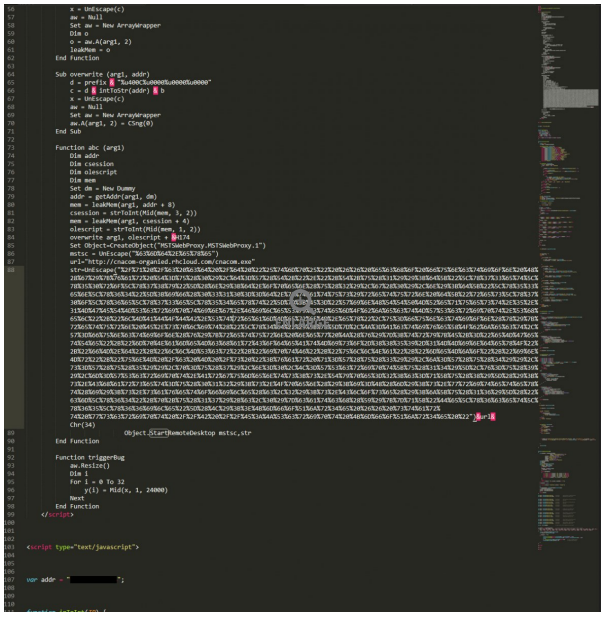

Figure 3 - A VBScript function named "abc" uses a combination of CVE-2016-0189 as well as what appears to be CVE-2015-0116 to gain code execution outside of the Internet Explorer (IE) sandbox

Following the exploit code, things get a lot more interesting. The user's external IP address is stored as a string and an *ipToInt()* function is defined, followed by a set of subroutines to collect details from the user machine. The code gathers the OS version, browser name, version, and language setting, Flash and Java versions, installed Office version, and finally the raw User-Agent string from the browser. This is all sent to the RHCloud host via a GET request.

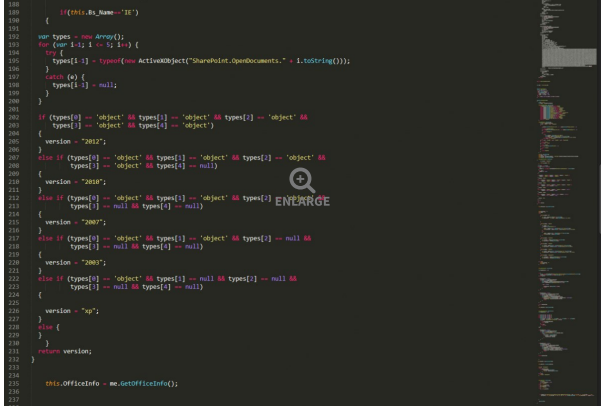

Figure 4 - The landing page collects many aspects of the user's platform, including MS Office version information

After the fingerprinting code, the user's IP address is checked against Taiwanese government network ranges. If the user is coming from one of the targeted networks and is using a version of Internet Explorer, exploitation will be attempted.

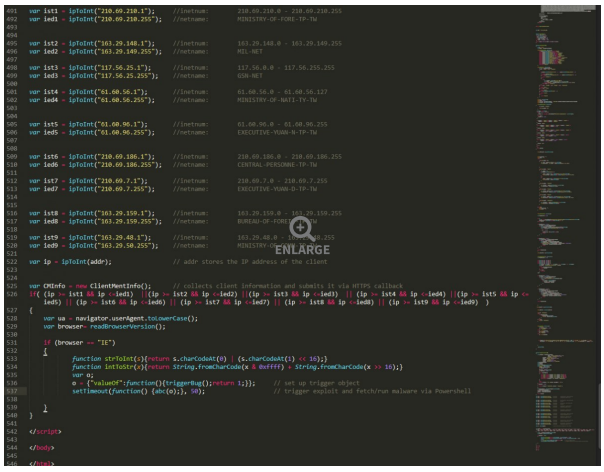

Figure 5 - The exploitation routine will be triggered for any Internet Explorer version, as long as the user's IP address is in one of the nine target networks

ThreatLabZ was able to follow the infection cycle and download a sample that appears to be a variant of the IXESHE AES malware. IXESHE is a family of backdoor malware known to be utilized by an attack group identified by various names including the IXESHE label, APT12, Numbered Panda, and DynCalc.

Figure 6 - **Upper**: among other changes seen, the new variant builds stack strings up to 4 bytes at a time. **Lower**: old variants do it byte-by-byte

Upon execution, the malware gathers the Windows username, hostname, local IP address, and Windows version. The hostname is fed to a PJW hash, or ElfHash function to generate a machine ID used in callbacks. The last step before initiating the C&C check-in is to achieve persistence by installing a run key in HKCU\Software\Microsoft\Windows\CurrentVersion\Run\Messenger.

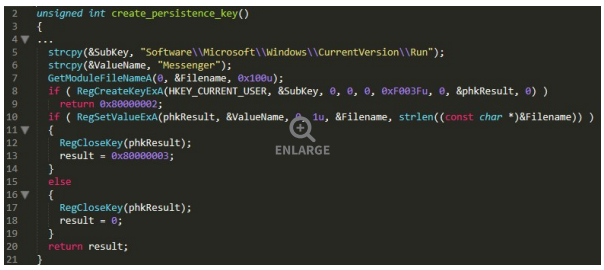

Figure 7 - Simplified decompiled code for the persistence mechanism shows the Run key utilized

This sample uses almost similar communication techniques as previous variants, with the addition of SSL. In our observations, we saw the server present a self-signed certificate with short, random-looking strings in the informational fields.

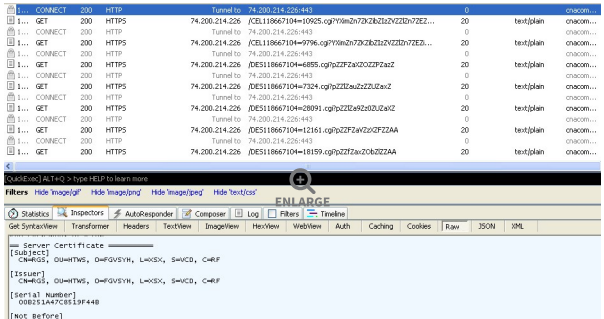

Figure 8 - A self signed certificate is used for the C&C server

Callback URLs

- /CEL%d=%d.cgi?%s - check-in at startup (and after certain C&C reset/error conditions)
- /DES%d=%d.cgi?%s - standard beacon, check for command
- /RES%d=%d.cgi?%s - response to rsh command
- /SDU%d=%d.cgi?%s - error response
- /SUS%d=%d.cgi?%s - check-out after receiving shutdown message

As can be seen above, the callback URLs utilize the same general format: three capital letters denoting the response function or condition, an integer representing the *PJW/ElfHash* based host ID, an equal sign ("="), a random integer, the string ".cgi?", and a base64 response blob (which in some cases simply encodes another random integer). The following regular expression matches this variant's URL path/query components: [CDRS][EDU][LSU]\d+=\d+\.cgi\?[a-zA-Z0-9=\+\/]+.

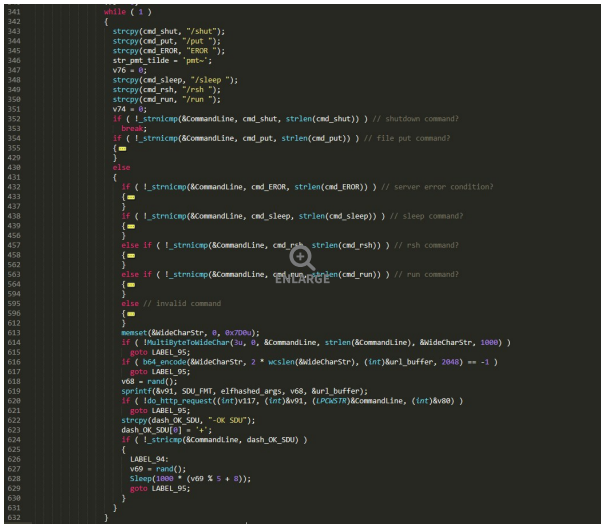

Figure 9 - A collapsed view of the decompiled C&C command processing code shows handling for multiple input commands and several response types

Unlike many historical IXESHE samples, it appears that this variant doesn't utilize campaign codes embedded in the malware itself. This may be due to a more centralized tracking system that only relies on the malware reporting a machine ID.

Conclusion

This analysis represents a snapshot of recent activity related to the CNACOM campaign. Additionally, we have identified an exploitation campaign active in August 2015 that appears to have utilized the HackingTeam Flash exploit for CVE-2015-5122, though the landing page at that time targeted a different set of Taiwanese government networks. Whether or not the threat actor behind this campaign is actually the group named APT12, the targeting of Taiwanese government networks and the similarity of this strain to historic IXESHE samples provide strong reasons for suspicion.

Zscaler ThreatlabZ will continue to monitor activity from this group ensuring protection against this threat.

Indicators of Compromise

Filename: cnacom.exe
Source: cnacom-organied.rhcloud\.com/cnacom.exe
MD5: ACFA9C664016BFE5DB92557E923744F0
Compile Time: 11/04/2016 11:56:27
Hardcoded User-Agent: Mozilla/5.0 (Windows NT 6.1; rv:43.0) Gecko/20100101 Firefox/43.0
C&C: 74.200.214.226

References

- http://www.trendmicro.com/cloud-content/us/pdfs/security-intelligence/white-papers/wp_ixeshe.pdf
- https://www.arbornetworks.com/blog/asert/wp-content/uploads/2014/06/ASERT-Threat-Intelligence-Brief-2014-07-Illuminating-Etumbot-APT.pdf

Keep in touch with us!

Email

Submit

By clicking the submit button, you are agreeing to our privacy policy.