Wireshark Tutorial: Examining Ursnif Infections

vireshark-tutorial-examining-ursnif-infections/

Brad Duncan

December 23, 2019

By Brad Duncan

December 23, 2019 at 6:00 AM

Category: Tutorial, Unit 42

Tags: pcap, Ursnif, Wireshark, Wireshark Tutorial

traffic-for-wireshark-column	-setup.pcap					
File Edit View Go C	apture Analyze	Sta	tistics Telephony Wirele	ss Tools Help		
	0 9 0 0			王		
http.request or ssl.handshake	type == 1				Expression	÷
Time	Dst	Port	Host	Server Name	Info	^
2018-08-03 19:06:20	192.0.79.32	80	college.usatoday.com		GET /2017/03/01/	
2018-08-03 19:06:20	192.0.78.19	443		r-login.wordpress.com	Client Hello	
2018-08-03 19:06:20	192.0.78.19	443		r-login.wordpress.com	Client Hello	
2018-08-03 19:06:20	192.0.77.32	443		s2.wp.com	Client Hello	
2018-08-03 19:06:20	1 2.0.77 32	443	гор.	s2. p.com	Client Hello	
2018-08-03 19:06:20	1 2.0.77 31	443	I U N	s2 p.com	Client Hello	
2018-08-03 19:06: <u>20</u>	192.0.77.32	443		s2.wp.com	Client Hello	
2018-08-03 19:06:20	192.0.77.32	443		s1.wp.com	Client Hello	
2018-08-03 19:06:20	192.0.77.32	443		s1.wp.com	Client Hello	
2018-08-03 19:06:20	192.0.77.32	443		s1.wp.com	Client Hello	
2018-08-03 19:06:20	216.58.218	443		fonts.googleapis.com	Client Hello	
2018-08-03 19:06:20	216.58.218	443		fonts.googleapis.com	Client Hello	
2018-08-03 19:06:20	52.84.125	80	d15krst4gi8g86.clou		GET /css/usatoda	
2018-08-03 19:06:20	52.84.125	80	d15krst4gi8g86.clou		GET /js/script.j	
2018-08-03 10-06-20	52 8/ 125	80	d15kpet/digage clou		GET /rec/usatoda	~~
🔵 🝸 traffic-for-wireshark-o	column-setup.pcap			Packets: 4448 · Displayed:	123 (2.8%) Profile: Defa	ult

This post is also available in: 日本語 (Japanese)

<u>Ursnif</u> is banking malware sometimes referred to as Gozi or IFSB. The Ursnif family of malware has been active for years, and current samples generate distinct traffic patterns.

This tutorial reviews packet captures (pcaps) of infection Ursnif traffic using <u>Wireshark</u>. Understanding these traffic patterns can be critical for security professionals when detecting and investigating Ursnif infections.

This tutorial covers the following:

Ursnif distribution methods

- Categories of Ursnif traffic
- Five examples of pcaps from Ursnif infections

Note: This tutorial assumes you have a basic knowledge of Wireshark, and it uses a customized column display shown in <u>this tutorial</u>. You should also have experience with Wireshark display filters as described in <u>this additional tutorial</u>.

Ursnif Distribution Methods

Ursnif can be distributed through web-based infection chains and malicious spam (malspam). In some cases, Ursnif is a follow-up infection caused by different malware families like <u>Hancitor</u>, as reported in <u>this recent example</u>.

We frequently find examples of Ursnif from malspam-based distribution campaigns, such as the example in Figure 1.



Figure 1. Flowchart from one of the more common Ursnif distribution campaigns.

Categories of Ursnif Traffic

This tutorial covers two categories of Ursnif infection traffic:

- Ursnif without HTTPS post-infection traffic
- Ursnif with HTTPS post-infection traffic

Malware samples from either of these categories create the same type of artifacts on an infected Windows host. For example, both types of Ursnif remain persistent on a Windows host by updating the Windows registry, such as the example shown in Figure 2.

ở Registry Editor				
File Edit View Favori	tes	Help		
📙 Printers	*	Name	Туре	Data
📕 Software		(Default)	REG_SZ	(value not set)
🖻 📙 Adobe	E	10236 [6FEDF42E-023C	REG_BI	ff 3a cf b9 71 ac d5 01
AppDataLow		(ACAED5F6-1B5	REG_BI	53 14 e3 fc 85 ac d5 01
A 📙 Software		ulli aecad Pnp	REG_BI	24 6b 65 6c 71 75 77 74 3d 22 63 67 6a 6d 6a 67 70 22 3b
Microsoft		AppIPQEC	REG_SZ	Fqm61v=new ActiveXObject('WScript.Shell');Fqm61v.Run('po
AAFBC		apssthci	REG_SZ	mshta "about: <hta:application><script></script></hta:application>

Figure 2. Example of Windows registry updates caused by samples of Ursnif, either with or without HTTPS post-infection traffic.

Example 1: Ursnif without HTTPS

The first pcap for this tutorial, *Ursnif-traffic-example-1.pcap*, is available <u>here</u>. The chain of events behind this traffic was tweeted <u>here</u>. Example 1 has been stripped of all traffic not directly related to the Ursnif infection.

Open the pcap in Wireshark and filter on *http.request* as shown in Figure 3.

		Ursnif	-traffic-example-1.pcap	+ _ E X
<u>F</u> ile <u>E</u> dit <u>V</u> iew <u>G</u> o <u>C</u> apture	Analyze Statistics	Telep	ohon <u>y W</u> ireless <u>T</u> o	ols <u>H</u> elp
	🤰 👵 🔶 🕭	k 🕅		् ⊞
http.request				Expression + basic basic+ basic+dns
Time	Dst	port	Host	Info
2019-12-06 20:13:09	8.208.24.139	80	w8.wensa.at	GET /api1/Gk5n2Z1f3uQNZRENzxY/BeR1PM_2
2019-12-06 20:13:10	8.208.24.139	80	8.208.24.139	GET /favicon.ico HTTP/1.1
2019-12-06 20:13:52	8.208.24.139	80	w8.wensa.at	GET /api1/zbEzPMStRCaZ9_2/F3evXmoHdrCu
2019-12-06 20:13:53	8.208.24.139	80	8.208.24.139	GET /favicon.ico HTTP/1.1
2019-12-06 20:13:55	8.208.24.139	80	w8.wensa.at	GET /api1/kKvyiA_2B_2BJs/yh7oiGH8Ye5A1
2019-12-06 20:14:03	8.208.24.139	80	api2.casus.at	GET /jvassets/xI/t64.dat HTTP/1.1
2019-12-06 20:14:13	8.208.24.139	80	h1.wensa.at	GET /api1/UuKEg2I0P1qjUnyuTAck/95ed4kt
2019-12-06 20:14:14	8.208.24.139	80	h1.wensa.at	POST /api1/ug78MmLblnoy2FXHqdHb/UVh2bv
2019-12-06 20:14:14	8.208.24.139	80	h1.wensa.at	GET /api1/5iM7TwwYCPMkV9cs0Dd/NPgCokz;
2019-12-06 20:14:17	8.208.24.139	80	h1.wensa.at	GET /api1/hT94MTx39Li/Nq7KAG9QNAYjBV/8
2019-12-06 20:14:18	8.208.24.139	80	h1.wensa.at	POST /api1/QIrpKXgnGrWnzmnWp5HKh/tBCNM
2019-12-06 20:19:14	8.208.24.139	80	h1.wensa.at	POST /api1/17tWIZ_2B/4Lq_2Bnl_2Fs8gjn_
2019-12-06 20:19:14	8.208.24.139	80	h1.wensa.at	POST /api1/XeqtPyhYWk/oJp_2F2IzcfLzGtj
2019-12-06 20:24:13	8.208.24.139	80	h1.wensa.at	GET /api1/tmUkcI5EDuJ1n/HQ5izsPA/MgnAF
2019-12-06 20:24:14	8.208.24.139	80	h1.wensa.at	POST /api1/8jtoAFvCdAmZ/QKr3sSP7QMk/71
2019-12-06 20:24:14	8.208.24.139	80	h1.wensa.at	POST /api1/z6rklzYLrAdXjwK/5Kp8_2BtgPk
2019-12-06 20:29:14	8.208.24.139	80	h1.wensa.at	POST /api1/jCiqa19rHxGg1H2LNJI/eq1VyC6
2019-12-06 20:29:15	8.208.24.139	80	h1.wensa.at	POST /api1/86JK8EKHjI/mlQVA4_2BBPewzto
2019-12-06 20:29:39	8.208.24.139	80	h1.wensa.at	POST /api1/gxB0y5dbM7T/sPCdITI8CZne3N/
2019-12-06 20:34:13	8 208 24 139	80	h1 wensa at	GET /ani1/wY4TvN]Ln6g/2t7TaGHZski1Dd/1
⊖ ℤ Request: Boolean				Packets: 1671 · Displayed: 25 (1.5%) Profile: Default

Figure 3. The pcap for example 1 filtered in Wireshark.

In this example, the Ursnif-infected host generates post-infection traffic to 8.208.24[.]139 using various domain names ending with .at. This category of Ursnif causes the following traffic:

- HTTP GET requests caused by the initial Ursnif binary
- HTTP GET request for follow-up data, with the URL ending in .dat
- HTTP GET and POST requests after Ursnif is persistent in the Windows registry

The following HTTP data is used during the traffic in our first example:

- Domain for initial GET requests: w8.wensa[.]at
- Request for follow-up data: hxxp://api2.casys[.]at/jvassets/xl/t64.dat
- Domain for GET and POST requests after Ursnif is persistent: h1.wensa[.]at

Follow the TCP stream for the very first HTTP GET request at 20:13:09 UTC. The TCP stream window shows the full URL. Note how the GET request starts with /api1/ and is followed by a long string of alpha-numeric characters with backslashes and underscores. Figure 4 highlights the GET request.

Wireshark · Follow TCP Stream (tc	p.stream eq 1) · Ursnif-traffic-example-1.pcap • _ = ×						
GET /api1/Gk5n2Z1f3uQNZRENzxY/BeRlPM_2FbyOTq4aK_2FIp/Dcf3zrOuZ613w/_2Fhq1ZS/ yoxosmdDTxp9Df8gT15FbeP/0sS35tjtdS/pG4Ea0Dugz9Ebe3MH/y5mC9bRgxZqo/fWI3ZuQUVgj/ JTT_2BCNRV147G/Sn40iLXUssnrQlch1AIfy/fVs5UZordH_2F4JX/VHN3v9rJEXrQciq/FioGSOAvDEVWsPDBsW/ hdabx0oBP/aFTak3JuZQkyahGQ2Dm6/G6fFPHEke56BRc_0A_0/D1H57FB2I6Y_2FZRwi1G8N/C4nUcv3 HTTP/1.1							
Accept: text/html, application/xhtml+xm Host: w8.wensa.at Accept-Language: en-US User-Agent: Mozilla/5.0 (Windows NT 6.1)	., */* : WOW64; Trident(7.0; rv:11.0) like Gecko						
Accept-Encoding: gzip, deflate							
DNT: 1 Connection: Keep-Alive	GET request after /api1/ has						
HTTP/1.1 200 OK Server: nginx Date: Fri, 06 Dec 2019 20:13:10 GMT	a long string of alpha-numeric						
Content-Type: text/html; charset=UTF-8							
Transfer-Encoding: chunked	and underscores						
Vary: Accept-Encoding							
Strict-Transport-Security: max-age=63072	2000; includeSubdomains						
Content-Type-Options: noshift Content-Encoding: gzip							
200a EPkl;;9h 22=Y.`pMfcem2.x.	8Z7W.moE\`BINF+.g. %d.d.gLkH2].)'s.ix.w]						
VX	*						
1 <mark>client</mark> pkt, 142 server pkts, 1 turn.							
Entire conversation (200 kB) -	Show and save data as ASCII - Stream 1 -						
Find:	Find <u>N</u> ext						
% Help	Filter Out This Stream Print Save as Back X Close						

Figure 4. Example of an HTTP GET request caused by our first Ursnif example.

We can find the same pattern from Ursnif activity caused by a Hancitor infection on December 10,2019. The pcap is available <u>here</u>. Mixed with the other malware activity, this December 10th example contains the following indicators for Ursnif:

- Domain for initial GET requests: foo.fulldin[.]at
- Request for follow-up data: hxxp://one.ahah100[.]at/jvassets/o1/s64.dat
- Domain for GET and POST requests after Ursnif is persistent: api.ahah100[.]at

Note how patterns from Ursnif traffic in the December 10th example are similar to the patterns we find in example 1. These patterns are commonly seen from Ursnif samples that do not use HTTPS traffic.

Example 2: Ursnif with HTTPS

The second pcap for this tutorial, *Ursnif-traffic-example-2.pcap*, is available <u>here</u>. Like our first pcap, this one has also been stripped of any traffic not related to the Ursnif infection.

Open the pcap in Wireshark and filter on *http.request or ssl.handshake.type* == 1 as shown in Figure 5. If you are using Wireshark 3.0 or newer, filter on *http.request or tls.handshake.type* == 1 for the correct results.

		Ursnif-traffic-example-2.pcap	+ _ E ×
<u>F</u> ile <u>E</u> dit <u>V</u> iew <u>G</u> o	o <u>C</u> apture <u>A</u> nalyze <u>S</u> tatisti	cs Telephony <u>W</u> ireless <u>T</u> ools <u>H</u> elp	
) 🛅 🖹 🎑 🗔 🧔 🕯) k 🖏 📃 📃 🔍 Q Q 🏢	
http.request or ss	l.handshake.type == 1		Expression + basic basic+ basic+dns
Time	Dst	port Host	Info
→ 2019-12-04 1	8:44 185.188.182.76	80 ghinatronx.com	GET /edgron/siloft.php?l=yorght6.cab
2019-12-04 1	8:46 45.143.93.81	80 bjanicki.com	GET /images/i8hvXkM_2F40/bgi3onEOH_2/
2019-12-04 1	8:46 45.143.93.81	80 bjanicki.com	GET /favicon.ico HTTP/1.1
2019-12-04 1	.8:46 45.143.93.81	80 bjanicki.com	GET /images/6a7GzE2PovJhysjaQ/HULhiLB
2019-12-04 1	.8:46 45.143.93.81	80 bjanicki.com	GET /images/aiXIa28QV6duat/PF_2BY9stc
2019-12-04 1	8:47 194.61.1.178	443 prodrigo29lbkf20.com	Client Hello
2019-12-04 1	8:48 194.61.1.178	443 prodrigo29lbkf20.com	Client Hello
2019-12-04 1	8:52 194.61.1.178	443 prodrigo29lbkf20.com	Client Hello
2019-12-04 1	8:57 194.61.1.178	443 prodrigo29lbkf20.com	Client Hello
2019-12-04 1	9:02 194.61.1.178	443 prodrigo29lbkf20.com	Client Hello
2019-12-04 1	9:07 194.61.1.178	443 prodrigo29lbkf20.com	Client Hello
2019-12-04 1	9:08 194.61.1.178	443 prodrigo29lbkf20.com	Client Hello
2019-12-04 1	9:13 194.61.1.178	443 prodrigo29lbkf20.com	Client Hello
2019-12-04 1	9:18 194.61.1.178	443 prodrigo29lbkf20.com	Client Hello
2019-12-04 1	9:19. 194.61.1.178	443 prodrigo29lbkf20.com	Client Hello
- Eramo 6: 296	bytos on wire (2000	hits) 286 bytes captured	(2000 hitc)
Ethornot TT	Src: HowlettP 10:47:	(00,00,00,00,00,00,00,00,00,00,00,00,00,	(3000 DILS)
Thernet Prot	ocol Version 4. Src	160, 192, 4, 101, Dst 185, 18	88.182.76
			· · · · · · · · · · · · · · · · · · ·
0000 20 e5 2a	b6 93 f1 00 08 02 1	c 47 ae 08 00 45 00 ·*···	••••••G••••E•
😑 💈 Ursnif-traffic-	-example-2.pcap		Packets: 1444 · Displayed: 52 (3.6%) Profile: Default

Figure 5. The pcap for our second example filtered in Wireshark.

This example has the following sequence of events:

- HTTP GET request that returns an initial Ursnif binary
- HTTP GET requests caused by the initial Ursnif binary
- HTTPS traffic after Ursnif is persistent in the Windows registry

Follow the TCP stream for the first HTTP GET request to ghinatronx[.]com. This TCP stream reveals a Windows executable or DLL file as shown in Figure 6. We can export the Ursnif binary from the pcap as described in this previous tutorial.

Wireshark · Follow TCP Stream (tcp.stream eq 0) · Ursnif-traffic-example-2.pcap	+ - = ×
GET /edgron/siloft.php?l=yorght6.cab HTTP/1.1 Accept: */* Accept: Encoding: gzip, deflate User-Agent: Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 6.1; W0W64; Trident/7.0; SLCC2; 2.0.50727; .NET CLR 3.5.30729; .NET CLR 3.0.30729; Media Center PC 6.0; .NET4.0C; .NET4.0C Host: ghinatronx.com Connection: Keep-Alive HTTP/1.1 200 OK Date: Wed, 04 Dec 2019 18:44:45 GMT Server: Apache/2.2.15 (CentOS) X-Powered-By: PHP/7.2.25 Content-Description: File Transfer Content-Disposition: attachment; filename="yorght6.cab" Expires: 0 Cache-Control: must-revalidate Pragma: public Content-Length: 290304 Connection: close Content-Type: apulication/octet-stream	A DLL
Most EXE or DLL files	*
1 client pkt, 201 server pkts, 1 turn.	
Entire conversation (290 kB) Show and save data as ASCII	Stream 0 🗘
Find:	Find <u>N</u> ext
Filter Out This Stream Print Save as B	ack X <u>C</u> lose

Figure 6. The first HTTP GET request returning a binary for Ursnif.

The next four HTTP requests to bjanicki[.]com were caused by the Ursnif binary. Follow the TCP stream for the first HTTP GET request to bjanicki[.]com at 18:46:21 UTC. This TCP stream shows the full URL. Note how the GET request starts with /images/ and is followed by a long string of alpha-numeric characters with backslashes and underscores before ending with .avi. This URL pattern is somewhat similar to Ursnif traffic from our first pcap. Figure 7 highlights a GET request from our second pcap.

Wireshark · Follow TCP Stream (tcp.stream eq 1) · Ursnif-traffic-example-2.pcap					
GET /images/i8hvXkM_2F40/bgi3onEOH_2/Fp_ wFLVNn_2By_2Fb2/WPHYci0rdY2dogSODh/YnkcD sP_2BvAuw9PjX/ugTn.avi HTTP/1.1	2FNWip7iwXT/I9ec6aw1_2BGhXbPixQHw/P7LK5Q_2F RKqk/sQG3_2BH_2FAoIu48Zkg/4rkH7uEXf_2FnP0QxI	t0TxcvC/ ← <h <="" td=""></h>			
Accept: text/html, application/xhtml+xml	, */*				
User-Agent: Mozilla/5.0 (Windows NT 6.1;	WOW64; Trident/7.0; rv:11.0) like Gecko				
Accept-Encoding: gzip, deflate					
Host: bjanicki.com	GET request after /Images	/ nas			
DNT: 1 Connection: Keep-Alive	a long string of alphanur	neric			
HTTP/1.1 200 OK	characters with backslashes	and			
Date: Wed, 04 Dec 2019 18:46:16 GMT					
Server: Apache/2.4.6 (CentOS) PHP/5.4.16	underscores before er	naing			
Set-Cookie: PHPSESSID=58k9nfnv1aahci3ss3	rtu6com3: nath=/: domain=.bianicki.com				
Expires: Thu, 19 Nov 1981 08:52:00 GMT		.avi			
Cache-Control: no-store, no-cache, must-	revalidate, post-check=0, pre-check=0				
Pragma: no-cache					
Set-Cookie: lang=en; expires=Fri, 03-Jan	-2020 18:46:16 GMT; path=/; domain=.bjanick:	i.com			
Connection: Keen-Alive					
Transfer-Encoding: chunked					
Content-Type: text/html; charset=UTF-8					
36314					
AmWPTSNZzZNuc0FuNsawbo6yM0lbaJ9fYozQ0hnX 4wJ5xznA4tQgWWhe9tMBGHKSXupBm/RZfmTIXL8x	7nkPcdmpM3kq+QmCc004HNOfvfhXew7U8o3AsvDMBhc/ JILMicxiNbRe9kBYhNRescQ/	PvsL4hc59			
3 client pkts, 160 server pkts, 5 turns.					
Entire conversation (232 kB) -	Show and save data as ASCII	Stream 1 🗘			
Find:		Find <u>N</u> ext			
%Help	Filter Out This Stream Print Save as Bac	k X <u>C</u> lose			

Figure 7. Example of an HTTP GET request from our second Ursnif example.

Unlike our first example, Ursnif in this second pcap generates HTTPS traffic after it becomes persistent on an infected Windows host. Use your *basic* web filter as described in this previous tutorial for a quick review of the HTTPS traffic. Note the HTTPS traffic to prodrigo29lbkf20[.]com as shown in Figure 8.

		Ursnif-traffic-example-2.pcap	+ - • ×
<u>File Edit V</u> iew	<u>Go</u> <u>Capture</u> <u>A</u> nalyze <u>S</u> tatis	itics Telephony <u>W</u> ireless <u>T</u> ools <u>H</u>	elp
	🗎 🗋 🖹 🎑 🗟 🗢 '	◈ ୲ୡ 📃 📃 🤍 ୦、 ୦、 🎚	
(http.request o	or ssl.handshake.type == 1) an	nd !(ssdp)	Expression + basic basic+ basic+dns
Time	Dst	port Host	Info
→ 2019-12-04	4 18:44 185.188.182.76	6 80 ghinatronx.com	GET /edgron/siloft.php?l=yorght6.cab
2019-12-04	4 18:4645.143.93.81	80 bjanicki.com	GET /images/i8hvXkM_2F40/bgi3onEOH_2/
2019-12-04	4 18:4645.143.93.81	80 bjanicki.com	GET /favicon.ico HTTP/1.1
2019-12-04	4 18:4645.143.93.81	80 bjanicki.com	GET /images/6a7GzE2PovJhysjaQ/HULhiLE
2019-12-04	4 18:4645.143.93.81	80 bjanicki.com	GET /images/aiXIa28QV6duat/PF_2BY9sto
2019-12-04	4 18:47194.61.1.178	443 prodrigo29lbkf20.com	Client Hello
2019-12-04	4 18:48194.61.1.178	443 prodrigo29lbkf20.com	n Client Hello 🗖 🛛 🏹 🖌
2019-12-04	4 18:52194.61.1.178	443 prodrigo29lbkf20.com	n Client Hello
2019-12-04	4 18:57194.61.1.178	443 prodrigo29lbkf20.com	n Client Hello 🔶 โลโนโด
2019-12-04	4 19:02194.61.1.178	443 prodrigo29lbkf20.com	n Client Hello
2019-12-04	4 19:07194.61.1.178	443 prodrigo29lbkf20.com	1 Client Hello CAUSED DV
2019-12-04	4 19:08194.61.1.178	443 prodrigo29lbkf20.com	n Client Hello
2019-12-04	4 19:13194.61.1.178	443 prodrigo29lbkf20.com	n Client Hello Ursnit
2019-12-04	4 19:18 194.61.1.178	443 prodrigo29lbkf20.com	n Client Hello
2019-12-04	4 19:19194.61.1.178	443_prodrigo29lbkf20.com	Client Hello
Eromo 61 2	PE bytes on wire (2000	hite) 286 bytes contures	1 (2000 bitc)
Ethornot T	T Src: Howlette 10:47	(00,00,00)	(3000 DICS)
, Internet D	rotocol Version 4 Sro	. 160 102 / 101 Det 195	100 100 76
- · ·			.100.102.70
0000 20 e5	2a b6 93 f1 00 08 02 :	1c 47 ae 08 00 45 00 ·*·	••••••••••••••••••••••••••••••••••••••
🔾 🍸 Ursnif-tra	ffic-example-2.pcap		Packets: 1444 · Displayed: 52 (3.6%) Profile: Default

Figure 8. Filtering on web traffic in Wireshark, highlighting the HTTPS traffic generated by Ursnif.

HTTPS traffic generated by this Ursnif variant reveals distinct characteristics in certificates used to establish encrypted communications. To get a closer look, filter on *ssl.handshake.type* == 11 (or *tls.handshake.type* == 11 in Wireshark 3.0 or newer). Select the first frame in the results and go to the frame details window. There we can expand lines and work our way to the certificate issuer data. Figure 9 shows how to begin.

	Ursnif-traffic-example-2.pcap							
<u>F</u> ile <u>E</u> dit <u>V</u> iew <u>G</u> o <u>C</u> apture <u>A</u> nalyze <u>S</u> tat	istics T	elephony <u>W</u> ireless	<u>T</u> ools <u>H</u> e	p				
🥖 🔳 🖉 😂 🚞 🖀 🖉 🍇 🔶	≫ 除		Q Q 🎹					
ssl.handshake.type eq 11 Expression + basic basic+ basic+dns								
Time Src	port	Dst	port	Info				
2019-12-04 18:47 194.61.1.178	443	10.12.4.101	49169	Server Hello, Certificate, Server Key Exc				
2019-12-04 18:48 194.61.1.178	443	10.12.4.101	49171	Server Hello, Certificate, Server Key Exc				
2019-12-04 18:52 194.61.1.178	443	10.12.4.101	49173	Server Hello, Certificate, Server Key Exc				
2019-12-04 18:57 194.61.1.178	443	10.12.4.101	49175	Server Hello, Certificate, Server Key Exc				
2019-12-04 19:02 194.61.1.178	443	10.12.4.101	49178	Server Hello, Certificate, Server Key Exc				
2019-12-04 19:07 194.61.1.178	443	10.12.4.101	49180	Server Hello, Certificate, Server Key Exc				
2019-12-04 19:08 194.61.1.178	443	10.12.4.101	49182	Server Hello, Certificate, Server Key Exc				
2019-12-04 19:13 194.61.1.178	443	10.12.4.101	49183	Server Hello, Certificate, Server Key Exc				
2019-12-04 19:18 194.61.1.178	443	10.12.4.101	49185	Server Hello, Certificate, Server Key Exc				
2019-12-04 19:19 194.61.1.178	443	10.12.4.101	49186	Server Hello, Certificate, Server Key Exc				
1 0010 10 01 10:01 101 01 1 170	440	10 10 1 101	40100	Conver Holle Contificate Conver Kov Eve				
 Frame 739: 994 bytes on wire (7952 bits), 994 bytes captured (7952 bits) Ethernet II, Src: Netgear_b6:93:f1 (20:e5:2a:b6:93:f1), Dst: HewlettP_1c:47:ae (00:08:02:1c:47:ae) Internet Protocol Version 4, Src: 194.61.1.178, Dst: 10.12.4.101 Transmission Control Protocol, Src Port: 443, Dst Port: 49169, Seq: 1, Ack: 163, Len: 940 Secure Sockets Layer A expand this line TLSV1.2 Record Layer: Handshake Protocol: Server Hello TLSV1.2 Record Layer: Handshake Protocol: Certificate then expand this line Content Type: Handshake (22) Version: TLS 1.2 (0x0303) Length: 622 Handshake Protocol: Certificate (11) Length: 618 Certificates Length: 615 								
0000 00 08 02 1c 47 ae 20 e5 2a	b6 93	f1 08 00 45 0	06	· · *····E·				
0010 03 d4 90 ca 00 00 80 06 d3	f9 c2	3d 01 b2 0a 0	C					
○ ℤ Ursnif-traffic-example-2.pcap				Packets: 1444 · Displayed: 47 (3.3%) Profile: Default				

Figure 9. Finding our way to the certificate issuer data.

As shown in Figure 9, we expand the line for **Secure Sockets Layer** in the frame details window. For Wireshark 3.0, this line shows as **Transport Layer Security**. Then we expand the line labeled **TLSv1.2 Record Layer: Handshake Protocol: Certificate**. Then we expand the line labeled **Handshake Protocol: Certificate**.

We keep expanding, until we find our way to the certificate issuer data as shown in Figure 10.

File	Edit View	Go Capture Apalyze Stati	istics T	orsnii-tramc-exa	mple-2.pcap : Tools Ho	In				
						ib				
			≫ №		a a 1					
SSI	.handshake.	type eq 11				X	Ex	pression + b	asic∣basic+∣b	asic+dns
Time		Src	port	Dst	port	Info				^
20	019-12-04	18:47 194.61.1.178	443	10.12.4.101	49169	Server	Hello,	Certificate,	Server Key	Exc
20	019-12-04	18:48 194.61.1.178	443	10.12.4.101	49171	Server	Hello,	Certificate,	Server Key	Exc
1 20	010_12_0/	10,50 10/ 61 1 170	112	10 10 / 101	10172	Corvor	H0110	Cortificato	Corvor Kou	EYC I
	- Handsha	ke Protocol: Certific	ate							
L -	Handsl	hake Type: Certificat	e (11)							
	Lengti	h: 618								
	Certi	ficates Length: 615								
	- Certi	ficates (615 bytes)								
	Cert	ificate Length: 612								
-	- Cert	ificate: 30820260308	201c9a	0030201020209	00c692c94	4106d77d	ifc ((id-at-common	Name=*,id-at-	organi
	- si	gnedCertificate								
		version: v3 (2)								
		serialNumber: 1430872	202472	34330108						
		signature (sha256With	nRSAEn	cryption)						
		issuer: rdnSequence ((0)							
		 rdnSequence: 6 item 	ıs (id∙	at-commonName	e=*,id-at	-organiz	zationa	lUnitName=1,i	d-at-organiza	ationNa
		RDNSequence item:	1 ite	m (id-at-coun	tryName=>	XX)				
		RDNSequence item:	1 ite	m (id-at-stat	eOrProvir	nceName=	:1)			_
		RDNSequence item:	1 ite	m (id-at-loca	lityName=	=1)				
		RDNSequence item:	1 ite	m (id-at-orga	nization	Name=1)				
		RDNSequence item:	1 ite	m (id-at-orga	nizationa	alUnitNa	ame=1)			
	RDNSequence item: 1 item (id-at-commonName=*)									
	+	validity								
	+	subject: rdnSequence	(0)							*
4										•
0000	00 08	02 1c 47 ae 20 e5 2a	b6 93	f1 08 00 45 0	00	G· · *··	···E·			
0010	03 d4	90 ca 00 00 80 06 d3	f9 c2	3d 01 b2 0a 0	с		= · · · ·			-
0 🛛	Ursnif-traf	fic-example-2.pcap				Pac	ckets: 144	14 · Displayed: 47	(3.3%) Profile:	Default

Figure 10. Certificate issuer data from HTTPS traffic caused by Ursnif.

In Figure 10 shown under the *Handshake Protocol: Certificate* line, we work our way down through the following items:

- Certificates (615 bytes)
- Certificate: 30820260308201c9a003020102020900c692c94106d77dfc...
- signedCertificate
- Issuer: rdnSequence (6)
- rdnSequence: 6 items (id-at-commonName=*,id-at-organizationalUnitN...

Individual items under the *rdnSequence* line show properties of the certificate issuer. These reveal the following characteristics:

- countryName=XX
- stateOrProvinceName=1
- localityName=1
- organizationName=1
- organizationalUnitName=1
- commonName=*

This issuer data is not valid, and these patterns are commonly seen in Ursnif infections. But what does legitimate certificate data look like? Figure 11 shows valid data from a certificate issued by DigiCert.

```
- Handshake Protocol: Certificate
   Handshake Type: Certificate (11)
   Length: 3926
   Certificates Length: 3923
 - Certificates (3923 bytes)
    Certificate Length: 1614
   - Certificate: 3082064a30820532a00302010202100bb0a4627238ee2dd6... (id-at-commonName=

    signedCertificate

        version: v3 (2)
        serialNumber: 0x0bb0a4627238ee2dd6e10b056dd23c2e
       signature (sha256WithRSAEncryption)
       - issuer: rdnSequence (0)
        rdnSequence: 3 items (id-at-commonName=DigiCert SHA2 Secure Server CA.id-at-or
          > RDNSequence item: 1 item (id-at-countryName=US)
          > RDNSequence item: 1 item (id-at-organizationName=DigiCert Inc)
          RDNSequence item: 1 item (id-at-commonName=DigiCert SHA2 Secure Server CA)
       validity
       subject: rdnSequence (0)
       subjectPublicKevInfo
```

Figure 11. Valid certificate issuer data.

One last thing about Ursnif is the IP address check by an Ursnif-infected host. This happens over DNS using a resolver at opendns[.]com. Like other IP address identifiers, this is a legitimate service. However, these services are commonly used by malware.

To see this IP address check, filter on *dns.qry.name contains opendns.com* and review the results as shown in Figure 12.



Figure 12. IP address check by an Ursnif-infected Windows host.

As shown in Figure 12, the Window host generated a dns query for resolver1.opendns[.]com followed by a DNS query to 208.67.222[.]222 for myip.opendns[.]com. The DNS query to myip.opendns[.]com returned the public IP address of the infected Windows host.

Example 3: Ursnif with Follow-up Malware

Our third pcap, *Ursnif-traffic-example-3.pcap*, is available <u>here</u>. This pcap also has unrelated activity stripped from the traffic, but it builds on our last example. Our third pcap includes what appears to be decoy traffic, and it also includes an HTTP GET request for follow-up malware. The sequence of events is:

- HTTP GET request that returns an initial Ursnif binary
- HTTP GET requests caused by the initial Ursnif binary, including decoy URLs
- HTTPS traffic after Ursnif is persistent in the Windows registry
- HTTP GET request for follow-up malware

Use your *basic* web filter as described in this previous tutorial for a quick review of the webbased traffic as shown in Figure 13.

		Ursnif-traffic-example-3.pcap			
<u>File Edit View Go</u> Capture Analy	yze <u>S</u> tatistics T	elephony <u>W</u> ireless <u>T</u> ools	Help		
	. 🔹 🔶 🗲 除				
(http.request or ssl.handshake.type	e == 1) and !(ssd	(p)	Expression + basic basic+ basic+dns		
Time Dst	port	Host	Info		
+ 2019-11-13 14:36 94.103	.94.88 80	sinicaleer.com	GET /zepoli/ironak.php?l=mateii8.cab H		
2019-11-13 14:36 172.21	7.1.142 80	google.com	GET /images/tR_2BBLJ5WGccm2Cvpr/7c0TGC		
2019-11-13 14:36 172.21	7.14.164 80	www.google.com	GET /images/errors/robot.png HTTP/1.1		
2019-11-13 14:36 172.21	7.14.164 80	www.google.com	GET /images/branding/googlelogo/1x/goo		
2019-11-13 14:37 194.1.2	236.191 80	ghdy656262oe.com	GET /images/bUU6H0I79/hRzOnZPYPhWfU2Sk		
2019-11-13 14:38 216.58	.194.142 80	google.com	GET /images/h4C8XQwC/Z7d0qPw0oy9yGmBRvl		
2019-11-13 14:39 216.58	.194.69 443	gmail.com	Client Hello		
2019-11-13 14:39 216.58	.194.69 443	gmail.com	Client Hello		
2019-11-13 14:39 216.58	.193.132 443	www.google.com	Client Hello		
2019-11-13 14:39 216.58	.193.132 443	www.google.com	Client Hello		
2019-11-13 14:39 194.1.2	236.191 80	ghdy656262oe.com	GET /images/CodzxTtiCg4nrPiJ51ry/uxx1fl _		
2019-11-13 14:41 172.21	7.14.174 80	google.com	GET /images/WJ30pCZsE_2Fi/LP7P17Hw/o_2I		
2019-11-13 14:41 216.58	.194.69 443	gmail.com	Client Hello		
2019-11-13 14:41 216.58	.194.69 443	gmail.com	Client Hello		
2019-11-13 14:41 194.1.2	236.191 80	ghdy656262oe.com	GET /images/qMY4GHlb/4hvJbIBd5m3ZUaqnX		
2019-11-13 14:41 194.1.2	236.191 80	ghdy656262oe.com	GET /favicon.ico HTTP/1.1		
2019-11-13 14:41 194.1.2	236.191 80	ghdy656262oe.com	GET /images/i3i00QaV0382xXNupBTPME/vSC		
2019-11-13 14:41 194.1.2	236.191 80	ghdy656262oe.com	GET /images/fbF27qOoXF/9Nlp42ISAcE_2BL		
2019-11-13 14:41 194.1.2	236.191 80	ghdy656262oe.com	GET /images/fbF27qOoXF/9Nlp42ISAcE_2BL		
2019-11-13 14:43 172.21	7.1.238 443	google.com	Client Hello		
	· · · ·		•		
Frame 6: 386 bytes on wir	e (3088 bits), 386 bytes captu	rea (3088 bits)		
Ethernet II, Src: HewlettP_1c:47:ae (00:08:02:1c:47:ae), Dst: Netgear_b6:93:f1 (20:e5:2a:b6:93:f1)					
Internet protocol Version	a crai 10				
0000 20 e5 2a b6 93 f1 00	08 02 1c 47	ae 08 00 45 00	• * • • • • • • • • • • • • • • • • • •		
○ 2 Ursnif-traffic-example-3.pcap			Packets: 1864 · Displayed: 27 (1.4%) Profile: Default		

Figure 13. Filtering our third pcap for web traffic in Wireshark.

In Figure 13, the initial HTTP request to sinicaleer[.]com returned a Windows executable for Ursnif. The remaining traffic visible Figure 13 was caused by the Ursnif executable until it became persistent.

Three HTTP requests to google[.]com follow similar URL patterns as Ursnif traffic to an actual malicious domain of ghdy656262oe[.]com. These HTTP GET requests to google[.]com appear to be decoy traffic, because they do not assist the infection. HTTPS traffic over TCP port 443 to gmail[.]com and www.google[.]com also serves no direct purpose for the infection, and that activity could also be classified as decoy traffic. Figure 14 shows an example of the decoy HTTP GET requests to google[.]com.

Wireshark · Follow TCP Stream (tcp.st	ream eq 1) · Ursnif-traffic-example-3.pcap 🔶 👝 🗆 🗙
GET /images/tR_2BBLJ5WGccm2Cvpr/7c0TGCarapE 0rQo1mpBWMzMhGYaWP5cGba/6cKKPu7M4k/upkGRgHt Yg85BsECNX/NAhr9.avi HTTP/1.1	DI01fwF90LUt/zBeIa00d1csb6/mKuc06Xs/ :jyo_2FnIA/YtgX2e1eVWz4/_2FIsETqwUK/i_2BnMfk/
Accept: text/html, application/xhtml+xml, * Accept-Language: en-US	
User-Agent: Mozilla/5.0 (Windows NT 6.1; WC	W64; Trident/7.0; rv:11.0) like Gecko
Accept-Encoding: gzip, deflate Decoy	URL to google.com where the
Connection: Keep-Alive GET	request after / <i>images</i> / has a
HTTP/1.1 404 Not Found Content-Type: text/html; charset=UTF-8	long string of alphanumeric
Referrer-Policy: no-referrer Content-Length: 1741	characters and backslashes
Date: Wed, 13 Nov 2019 14:36:54 GMT	before ending with <i>.avi</i>
html	•
<ntml lang="en"></ntml>	
<pre><meta charset-utrop<br=""/><meta content="initial-scal
<title>Error 404 (Not Found)!!1</title></pre></td><td>e=1, minimum-scale=1, width=device-width" name="viewport"/></pre>	
<style></style>	

Figure 14. Decoy HTTP GET request by the Ursnif-infected host to a Google domain.

Note the HTTP traffic to ghdy656262oe[.]com. The first two GET requests to ghdy656262oe[.]com return a *404 Not Found* response as shown in Figure 15. The third HTTP GET request returns a *200 OK* response, and the infection continues as shown in Figure 16.

Wireshark · Follow TCP Stream (tcp.stream eq 4) · Ur	snif-traffic-example-3.pcap 🔶 💷 🗙
GET /images/bUU6H0I79/hRzOnZPYPhWfU2SkId8V/6Zbr_2Bea s5X1Qxv0NqxC9/QEVpggo6/o16QuDm_2BCurGX_2BOffoS/n2ht9 22_2F_2B/S4_2F.avi HTTP/1.1 Accept: text/html, application/xhtml+xml, */* Accept-Language: en-US User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64; Tric Accept-Encoding: gzip, deflate Host: ghdy6562620e.com HTTPPGET DNT: 1 Connection: Keep-Alive	alYF3V_2FDc/ZAfMyD0JzjmEwBbAWx0XFW/ DtQDCk/sRlH9Fz_2FmERcqTo/TMgKvSkI6vFP/ dent/7.0; rv:11.0) like Gecko request to ghdy656262.com
HTTP/1.0 404 Not Found Date: Wed, 13 Nov 2019 14:37:37 GMT Server: Apache/2.4.6 (CentOS) PHP/5.4.16 X-Powered-By: PHP/5.4.16 Set-Cookie: PHPSESSID=17cer7uotsdfksqkr27mdvu2k3; pa Expires: Thu, 19 Nov 1981 08:52:00 GMT Cache-Control: no-store, no-cache, must-revalidate, Pragma: no-cache Set-Cookie: lang=en; expires=Fri, 13-Dec-2019 14:37: Content-Length: 0 Connection: close Content-Type: text/html; charset=UTF-8	Found ath=/; domain=.ghdy6562620e.com post-check=0, pre-check=0 :37 GMT; path=/; domain=.ghdy6562620e.com
1 client pkt, 1 server pkt, 1 turn. Entire conversation (971 bytes) Find: Find: Filter Out	ve data as ASCII Stream 4 Find Next This Stream Print Save as Back XClose

Figure 15. First two HTTP GET requests to malicious Ursnif domain return a 404 Not Found response.

		Ursnif-traffic-example-3.pca	q
<u>F</u> ile <u>E</u> dit <u>V</u> iew <u>G</u> o <u>C</u> apture <u>A</u> nalyze	Statistics Te	elephony <u>W</u> ireless <u>T</u> ools	s <u>H</u> elp
	🔶 📎 🏟		
(http.request or ssl.handshake.type ==	1) and !(ssd	Expression + basic basic+ basic+dns	
Time Dst	port	Host	Info
→ 2019-11-13 14:36… 94.103.94.	88 80	sinicaleer.com	GET /zepoli/ironak.php?l=mateii8.cab HTTP,
2019-1-12-14:20 172-017-1	¹ 42 80	google.com	GET /images/tR_2BBLJ5WGccm2Cvpr/7c0TGCara
²⁰¹⁹⁻¹ 404 Not Found	164 80	www.google.com	GET /images/errors/robot.png HTTP/1.1
2019-1	10. 80	www.google.com	GET /images/branding/googlelogo/1x/googlel
2019-11-13 14:37 194.1.236	191 8	ghdy656262oe.com	GET /images/bUU6H0I79/hRzOnZPYPhWfU2SkId8
2019-11-13 14:38 216.58.194	1.142 80	google.com	GET /images/h4C8XQwC/Z7d0qPw0oy9yGmBRvbKh;
2019-11-13 14:39 216.58.194	1.69 443	gmail.com	Client Hello
2019-14 10 14.00 010 F0 10	69 443	gmail.com	Client Hello
²⁰¹⁹⁻]404 Not Found	132 443	www.google.com	Client Hello
2019-1	13. 448	www.google.com	Client Hello
2019-11-13 14:39 194.1.236	191 8	ghdy656262oe.com	GET /images/CodzxTtiCg4nrPiJ51ry/uxx1fhPti
2019-11-13 14:44 170 017 1	174 80	google.com	GET /images/WJ30pCZsE_2F1/LP7P17HW/0_2BdW
2019-11-13 14:4 200 OK	69 443	gmail.com	Client Hello
2019-11-13 14:4	69 443	gmail.com	Client Hello
2019-11-13 14:41 194.1.236.	191 8	gndy6562620e.com	GET /Images/qMY4GHID/4nVJDIBd5m3ZUaqnXZ4nr
2019-11-13 14:41 194.1.236.	191 80	gndy6562620e.com	GET /Tavicon.ico HTTP/1.1
2019-11-13 14:41 194.1.236.	191 80	gndy6562620e.com	GET /images/13100QaV0382XXNUpBTPME/VSCSW56
2019-11-13 14:41 194.1.236.	191 80	gnay6562620e.com	GET /images/fbF2/quoxF/9N1p421SACE_2BLWS/1
2019-11-13 14:41 194.1.236.	191 80	gndy6562620e.com	GET /Images/TDF2/qU0XF/9N1p421SACE_2BLWS/T
2019-11-13 14:43 172.217.1.	238 443	googie.com	Client Hello
Frame 6: 386 bytes on wire (3088 bits). 386 bytes captu	red (3088 bits)
> Ethernet II. Src: HewlettP 1	c:47:ae (00:08:02:1c:47:ae)	. Dst: Netgear b6:93:f1 (20:e5:2a:b6:93:f1)
 Internet Protocol Version 4, 	Src: 10.	11.13.101, Dst: 94	.103.94.88
0000 20 e5 2a b6 93 f1 00 08	02 1c 47	ae 08 00 45 00	·*····································
😑 🌋 Ursnif-traffic-example-3.pcap			Packets: 1864 · Displayed: 27 (1.4%) Profile: Default

Figure 16. Some false starts before the Ursnif infection continues.

Since the first HTTP GET request to ghdy6562620e[.]com was not a 200 OK, the infected Windows host cycled through other malicious domains to continue the infection. These two domains are tnzf3380au[.]top and xijamaalj[.]com. However, the DNS queries for these domains returned a "No such name" in response, so the infected Windows host went back to trying ghdy6562620e[.]com.

Use the following Wireshark filter to better see this sequence of events:

((http.request or http.response) and ip.addr eq 194.1.236.191) or dns.qry.name contains tnzf3380au or dns.qry.name contains xijamaalj

The results should appear similar to the column display in Figure 17.

		Ur	snif-traffic-example-	3.pcap	+ _ = ×
File Edit Vie	w <u>Go C</u> apture <u>A</u> nalyze <u>S</u> tati	stics Tele	ephony Wireless	Tools	Help
		A 14 d		- -	Ŧ
		× 182 5		~ ~	
📕 ((http.reque	st or http.response) and ip.addr	eq 194.1.2	236.191) or dns.q	ry.nam	ne co⊠ 🖃 🝸 Expression + basic basic+ basic+dns
Time	Dst	port H	Host		Info
	13 14:37 194.1.236.191	80 g	ghdy6562620e.	com	GET /images/bUU6H0I79/hRzOnZPYPhWfU2SkId8V/6
- 2019-11-	13 14:37 10.11.13.101	49167			HTTP/1.0 404 Not Found
2019-11-	13 14:37 10.11.13.1	53			Standard query 0x2f94 A tnzf3380au.top
2019-11-	13 14:37 10.11.13.101	60214			Standard query response 0x2f94 No such name
2019-11-	13 14:38 10.11.13.1	53			Standard query 0x7871 A xijamaalj.com
2019-11-	13 14:38 10.11.13.101	55914			Standard query response 0x7871 No such name
2019-11-	13 14:39 194.1.236.191	80 (40175	gnay6562620e.	com	UTTD (1 0 404 Not Found
2019-11-	13 14:39 10.11.13.101	491/5	abdy65626200	com	GET /images/gMV/CH1b/4by1bTPd5m27UagpVz4pNb/
2019-11-	13 14.41 194.1.230.191 13 14.41 10 11 13 101	/9182	Jiiuy0302020e.	COIII	HTTP/1 1 200 OK (text/html)
2019-11-	13 14:41 194 1.236 191	80 0	nhdv6562620e	com	GET /favicon.ico HTTP/1.1
2019-11-	13 14:41 10.11.13.101	49182	gnay 000202000	00111	HTTP/1.1 200 OK (image/vnd.microsoft.icon)
2019-11-	13 14:41 194.1.236.191	80 0	ahdv6562620e.	com	GET /images/i3i000aV0382xXNupBTPME/vSCSw5eR
2019-11-	13 14:41 10.11.13.101	49183	5 5		HTTP/1.1 200 OK (text/html)
2019-11-	13 14:41 194.1.236.191	80 0	ghdy6562620e.	com	GET /images/fbF27qOoXF/9Nlp42ISAcE_2BLWS/nbz
2019-11-	13 14:41 194.1.236.191	80 0	ghdy6562620e.	com	GET /images/fbF27qOoXF/9Nlp42ISAcE_2BLWS/nbz
2019-11-	13 14:41 10.11.13.101	49181			HTTP/1.1 200 OK (text/html)
•					Þ
> Frame 612	2: 501 bytes on wire (4	308 bits	s), 501 bytes	s capt	tured (4008 bits)
→ Ethernet	II, Src: HewlettP_1c:4	7:ae (00	0:08:02:1c:47	7:ae),	, Dst: Netgear_b6:93:f1 (20:e5:2a:b6:93:f1)
4 Internet	Brotocol Version / Cr.	· 10 1·	1 12 101 Det	- 10/	
0000 20 e	5 2a b6 93 f1 00 08 02	1c 47 a	ae 08 00 45 00	0	• * • • • • • • • • • • • • • • • • • •
🔾 🗹 Ursnif-t	raffic-example-3.pcap				Packets: 1864 · Displayed: 17 (0.9%) Profile: Default
L					

Figure 17. Filtering to show how the infected Windows host tries Ursnif-related domains before it hits a 200 OK in HTTP traffic.

To review the rest of the infection, use your *basic* web filter and scroll to the end of the results. Figure 18 shows the post-infection traffic after Ursnif becomes persistent.

					Ursnif-traffic-example-3.pcap		+ - = ×			
<u>F</u> ile <u>E</u> o	lit <u>V</u> iew <u>(</u>	<u>Go</u> _apti	ure <u>A</u> nalyze <u>S</u>	statistics	elephony <u>W</u> ireless <u>T</u> ools <u>H</u> elp					
		🗎 🔝 🖹	🔇 🎑 🔍	🔶 🏵 候	😫 📃 🔍 Q Q 💵					
(http:	(http.request or ssl.handshake.type == 1) and !(ssdp) Expression + basic basic+ basic+dns									
Time			Dst	port	Host	Info	^			
201	9-11-13	14:39	216.58.193	.132 443	www.google.com	Client Hello				
201	9-11-13	14:39	194.1.236.3	191 80	ghdy656262oe.com	GET /images/CodzxTtiCg4nrPi	J51ry/			
201	9-11-13	14:41	172.217.14	.174 80	google.com	GET /images/WJ30pCZsE_2Fi/L	P7P17H			
201	9-11-13	14:41	216.58.194	.69 443	gmail.com	Client Hello				
201	9-11-13	14:41	216.58.194	.69 443	gmail.com	Client Hello				
201	9-11-13	14:41	194.1.236.3	191 80	ghdy656262oe.com	GET /images/qMY4GHlb/4hvJbI	Bd5m3Z			
201	9-11-13	14:41	194.1.236.3	191 80	ghdy656262oe.com	GET /favicon.ico HTTP/1.1				
201	9-11-13	14:41	194.1.236.3	191 80	ghdy656262oe.com	GET /images/i3i00QaV0382xXN	UPBTPN			
201	9-11-13	14:41	194.1.236.3	191 80	ghdy656262oe.com	GET /images/fbF27qOoXF/9Nlp	42ISA			
201	9-11-13	14:41	194.1.236.3	191 80	ghdy656262oe.com	GET /images/fbF27qOoXF/9Nlp	42ISAc			
201	9-11-13	14:43	172.217.1.2	238 443	google.com	Client Hello				
201	9-11-13	14:43	216.58.194	.69 443	gmail.com	Client Hello				
201	9-11-13	14:43	83.166.242	.144 443	vnt69tnjacynthe.com	Client Hello				
201	9-11-13	14:43	83.166.242	.144 443	vnt69tnjacynthe.com	Client Hello				
201	9-11-13	14:43	83.166.242	.144 443	vnt69tnjacynthe.com	Client Hello				
201	9-11-13	14:43	83.166.242	.144 443	vnt69tnjacynthe.com	Client Hello				
201	9-11-13	14:48	83.166.242	.144 443	vnt69tnjacynthe.com	Client Hello				
201	9-11-13	14:48	199.188.200	9.89 80	carresqautomotive.com	GET /jjwekr.rar HTTP/1.1				
4							•			
→ Fram	e 6: 380	6 bytes	on wire (3	8088 bit	s), 386 bytes captured ((3088 bits)	^			
→ Ethe	rnet II,	, Src:	HewlettP_1	::47:ae	(00:08:02:1c:47:ae), Dst	: Netgear_b6:93:f1 (20:e5:2a	:b6:93:f1)			
Thto	rnot Dr	ntocol	Vorcion 4	Sroy 10	11 12 101 Det. 04 102	01 00				
0000	20 65 2	a h6 93	3 f1 00 09	02 1c /	ae 08 00 45 00 .*···					
0000	20 65 2	.u 00 93	, 11 00 00	02 10 41	ue 66 66 45 66 · · · ·	5 · · E ·	•			
0 🛛 🛛	Jrsnif-traff	ic-examp	le-3.pcap		Pa	ackets: 1864 · Displayed: 27 (1.4%) Pro	ofile: Default			

Figure 18. Post-infection traffic after Ursnif becomes persistent on the victim's Windows host.

In Figure 18, after five HTTP GET requests to ghdy6562620e[.]com, we find traffic generated by the infected Windows host after Ursnif becomes persistent. This includes HTTPS traffic to google[.]com and gmail[.]com.

Traffic to vnt69tnjacynthe[.]com should have the same type of certificate issuer data we witnessed in our second pcap. But this traffic includes an HTTP GET request to carresqautomotive[.]com ending with .rar.

This URL ending in .rar returned follow-up malware. However, this follow-up malware is encoded or otherwise encrypted when sent over the network. The binary decoded on the infected Windows host, which is not seen in the infection traffic. Follow the TCP stream for the HTTP GET request to carresqautomotive[.]com, and you should see the same data as shown in Figure 19.

Wireshark · Follow TCP Stream (tcp.:	tream eq 24) · Ursnif-traffic-example-3.pcap 🔶 👝 🗉 🗙
GET /jjwekr.rar HTTP/1.1 User-Agent: Mozilla/4.0 (compatible; MSIE Host: carresqautomotive.com Connection: Keep-Alive Cache-Control: no-cache HTTP/1.1 200 OK Date: Wed, 13 Nov 2019 14:48:05 GMT Server: Apache Last-Modified: Wed, 13 Nov 2019 12:17:17	8.0; Windows NT 6.1; Win64; x64) Content type shows as lication/x-rar-compressed , but the binary is encoded or encrypted data and <u>not</u> a rar archive.
Accept-Ranges: bytes Content-Length: 196679 Content-Type: application/x-rar-compressed y.i.6s.be.m{bY m.8j.K.kp y69.Pk.4'.0X2xo. 3I.BA@LzwE^")B.0{a_1} 0LW1C~{.0.I0 Y=.S!.qcVk.GG=J/H.]RHU.=R=.0 q>.K89uua.)f5[i. \.^0W~.W.Yg'.T.R./X(3.E.	d a&+]KQ.1E:DPh.w./ Jt&59.,QgS1T.:." .j D.EqK\4,.yY .Bn4RJ.r}BTQ. >s1.kcydG e.wpa7. <r4vw. Bx.?.;. W. </r4vw.
.h. *;VW/e*i4 +>tW. 9N.1 4\$ * KHS *	.C7+^)ao%MG.d *%XO.a`aaaa
Find:	Find Next Filter Out This Stream Print Save as Back X Close

Figure 19. Follow-up malware sent to an Ursnif-infected Windows host.

This data is encrypted, so we cannot export a copy of the follow-up malware from the pcap. Therefore, we must rely on other post-infection traffic to determine what type of malware was sent to the Ursnif-infected host.

We have seen various types of follow-up malware from Ursnif infections, including <u>Dridex</u>, <u>IcedID</u>, <u>Nymain</u>, <u>Pushdo</u>, and <u>Trickbot</u>.

Our next example is an Ursnif infection with Dridex as the follow-up malware.

Example 4: Ursnif Infection with Dridex

Our fourth pcap, *Ursnif-traffic-example-4.pcap*, is available <u>here</u>. Unlike our first three examples, this pcap does not have unrelated activity stripped from the traffic.

Use your *basic* web filter to get a better idea of the traffic. Your results should appear similar to Figure 20.

			Ursnif-traffic-example-4.pcap	• - • ×						
File Edit Viev	v <u>G</u> o <u>C</u> apture <u>A</u> nalyze <u>S</u> tatist	ics Tele	ephony <u>W</u> ireless <u>T</u> ools <u>H</u> elp							
(http.request	or ssl.handshake.type == 1) and	Expression + basic basic+ basic+dns								
Time	Dst	port	Host	Info						
	2 14:53 104.124.60.139	80	www.msftncsi.com	GET /ncsi.txt HTTP/1.1						
2019-11-1	2 14:53 80.85.159.236	80	oklogallem.com	GET /zepoli/ironak.php?l=luntsu1.cab						
2019-11-1	2 14:54 172.217.2.238	80	google.com	GET /images/SPdsgBJ5WiV_2BAGp5Z/kN8cg						
2019-11-1	2 14:54 172.217.1.132	80	www.google.com	GET /images/errors/robot.png HTTP/1.1						
2019-11-1	2 14:54 172.217.1.132	80	www.google.com	GET /images/branding/googlelogo/1x/gc						
2019-11-1	2 14:54 72.21.81.200	443	r20swj13mr.microsoft	Client Hello						
2019-11-1	2 14:54 72.21.81.200	443	<pre>iecvlist.microsoft.com</pre>	Client Hello						
2019-11-1	2 14:54 72.21.81.200	443	r20swj13mr.microsoft	Client Hello						
2019-11-1	2 14:54 72.21.81.200	443	<pre>iecvlist.microsoft.com</pre>	Client Hello						
2019-11-1	2 14:55 194.87.147.244	80	kh2714ldb.com	GET /images/58HuD8Vcxh0H06K/eUWS28C7J						
2019-11-1	2 14:55 194.87.147.244	80	kh2714ldb.com	GET /favicon.ico HTTP/1.1						
2019-11-1	2 14:57 216.58.194.78	80	google.com	GET /images/z5FTh4xviE9DaSwTyQW/XXFK4						
2019-11-1	2 14:58 172.217.12.69	443	gmail.com	Client Hello						
2019-11-1	2 14:58 172.217.12.69	443	gmail.com	Client Hello						
2019-11-1	2 14:58 172.217.6.132	443	www.google.com	Client Hello						
2019-11-1	2 14:58 172.217.6.132	443	www.google.com	Client Hello						
2019-11-1	2 14:58 194,87,147,244	80	kh2714ldb.com	GET /images/KAD2ClWeS5zCvPqDjb/ 2FI0v						
2019-11-1	2 14:58 194.87.147.244	80	kh2714ldb.com	GET /images/4JdEdtEN/0T1uFTEBiZ5JTc 2						
2019-11-1	2 14:58 194.87.147.244	80	kh2714ldb.com	GET /images/2RvOtXHg00lg7sBop68xv/hxk						
2019-11-1	2 14:59 216.58.194.142	443	google.com	Client Hello						
2019-11-1	2 14:59 172,217,12,69	443	gmail.com	Client Hello						
4				•						
Frame 47:	151 bytes on wire (1208	bits), 151 bytes captured (1208 bits)						
Ethernet :	II, Src: HewlettP_1c:47:	ae (0	9:08:02:1c:47:ae), Dst:	Netgear_b6:93:f1 (20:e5:2a:b6:93:f1)						
Internet	Protocol Version 4, Src:	10.1	1.12.101, Dst: 104.124.	60.139						
→ Transmiss	ion Control Protocol. Sr	c Por	t: 49157. Dst Port: 80.	Sea: 1. Ack: 1. Len: 97						
0000 20 e5	2a b6 93 f1 00 08 02 1	c 47 a	e 08 00 45 00 ·*····	• • · G · · · E ·						
0010 00 89	00 47 40 00 80 06 3e b	1 0a 0	b 0c 65 68 7c ···G@··	• >••••eh						
😑 🗹 Ursnif-tr	affic-example-4.pcap			Packets: 2509 · Displayed: 37 (1.5%) Profile: Default						

Figure 20. Traffic from our fourth pcap filtered in Wireshark.

This pcap has the same sequence of events as our previous example, but it adds postinfection activity from the follow-up malware:

- HTTP GET request that returns an initial Ursnif binary
- HTTP GET requests caused by the initial Ursnif binary, including decoy URLs
- HTTPS traffic after Ursnif is persistent in the Windows registry
- HTTP GET request for follow-up malware
- Post-infection activity from the follow-up malware

In this fourth example, the HTTP GET request for an initial Ursnif binary is to oklogallem[.]com. Ursnif causes HTTP GET requests to kh2714ldb[.]com before the infection becomes persistent.

Figure 21 shows activity after Ursnif is persistent, where Ursnif causes HTTPS traffic to s9971kbjjessie[.]com. We then see an HTTP GET request to startuptshirt[.]my for the follow-up malware. Finally we find post-infection traffic caused by the follow-up malware.

	Ursnif-traffic-example-4.pcap + ×										
File E	dit <u>V</u> iew	<u>Go</u> <u>C</u> apt	ure <u>A</u> r	alyze S	Statisti	ics Te	elephony <u>W</u> ireless <u>T</u> ools	<u>H</u> elp			
		🗅 🗋 🔰	۵ (۵.	\$	>					
📕 (http	(http:request or ssl.handshake.type == 1) and !(ssdp)									\times	Expression + basic basic+ basic+dn
Time			Dst			port	Host	Info			
201	9-11-12	14:58	194.8	37.147	.244	80	kh2714ldb.com	GET	/ir	nages/K/	AD2ClWeS5zCvPqDjb/_2FIQy2xW/CNW4
201	9-11-12	14:58	194.8	37.147	.244	80	kh2714ldb.com	GET	/ir	nages/4.	JdEdtEN/QT1uFTEBiZ5JTc_2BOsC7EZ/
201	9-11-12	14:58	194.8	37.147	.244	80	kh2714ldb.com	GET	/ir	nages/2F	RvOtXHgQ0lq7sBop68xy/hxK_2F8EVdC.
201	9-11-12	14:59	216.5	58.194	.142	443	google.com	Clie	nt	Hello	
201	9-11-12	14:59	172.2	217.12	.69	443	gmail.com	Clie	nt	Hello	- HI IPS traffic after
201	9-11-12	14:59	85.14	43.219	.95	443	s9971kbjjessie.com	Clie	nt	Hello	Urenific pareictant
201	9-11-12	15:04	85.14	43.219	.95	443	s9971kbjjessie.com	Clie	nt	Hello	
201	9-11-12	15:09	85.14	43.219	.95	443	s9971kbjjessie.com	Clie	nt	Hello	
201	9-11-12	15:09	85.14	43.219	.95	443	S99/1KD]]essie.com	CIIe	ent	Hello	
201	9-11-12	15:09	124.2	217.25	5.96	80	startuptsnirt.my	GEI	/W	-conter	nt/uploads/2019/11/jjashdedw.rar
201	9-11-12	15:09	94.14	40.114	.0	443	Troffic coulord	Clie	ent.	Hello	
201	9-11-12	15:09	94.14	40.114	.0	443	Trainc caused	Clie	nt	Hello	
201	9-11-12	15.10	94.14	40.114	.0	443	by follow-up	Clie	nt	Hello	I URL for follow-up
201	9-11-12	15.10	5 61	34 51	.0	443	by follow-up	Clie	nt	Hello	
201	9-11-12	15:14	5.61	34.51		443	malware	Clie	nt	Hello	maiware
201	9-11-12	15:14	5.61	.34.51		443		Clie	nt	Hello	
201	9-11-12	15:14	5.61	.34.51		443		Clie	nt	Hello	
201	9-11-12	15:14	85.14	43.219	.95	443	s9971kbjjessie.com	Clie	nt	Hello	•
201	9-11-12	15:15	85.14	43.219	.95	443	s9971kbjjessie.com	Clie	nt	Hello	
201	9-11-12	15:15	85.14	43.219	. 95	443	s9971kbiiessie.com	Clie	nt	Hello	
		Fa huit			(4000	- I- 2 -	-) 454 bubs	and a	4.0		>
Fran	ne 47: 1	ST Dyte	s on	wire (1208	DIC	s), 151 bytes captu	rea (12	08 DICS) heinzift (201051201heinzift)
Ethe	ernet II	, SrC:	Verei	on 4	5:47: Sro	ae (11 12 101 Det: 104	DST:	E C	120	D0:93:11 (20:05:28:D6:93:11)
Tran	ernet Pr	n Contr		on 4,	SIC:	T0.	11.12.101, DSL: 104	. 124.	00	.139 ag: 1	Ack: 1 Len: 97
0000	20 e5 2	2a h6 93		0.020	02 1	c 47	ae 08 00 45 00	*		-G E -	AUK. 1, LEH. 3/
0010	00 89 0	00 47 40	000 8	30 06	3e h	1 0a	0h 0c 65 68 7c	· G@ · ·	. >	>···ehl	
07	Ursnif-traff	ic-examp	le-4.pc	ap	55 0	_ 54		00		Packets	s: 2509 · Displayed: 37 (1.5%) Profile: Default
	oronn dan	.e examp	.cpc	-P						. achece	

Figure 21. Activity from the infection after Ursnif is persistent.

Our fourth example follows the same infection patterns as our third pcap, but now we also have HTTPS/SSL/TLS traffic to 94.140.114[.]6 and 5.61.34[.]51 without any associated domain name. This is Dridex post-infection traffic.

Certificate issuer data for Dridex is different than certificate issuer data for Ursnif. Use the following filter to review the Dridex certificate data in our fourth pcap:

(ip.addr eq 94.140.114.6 or ip.addr eq 5.61.34.51) and ssl.handshake.type eq 11

Note: if you are using Wireshark 3.0 or newer, use *tls.handshake.type* instead of *ssl.handshake.type*.

Select the first frame in the results, go to the frame details window, and expand the certificate-related lines as shown by our second example in Figures 9 and 10. Examining certificate issuer data from our fourth pcap should look similar to Figures 22 and 23.

	Ursnif-traffic-example-4.pcap + _ 🗆 🗙									×											
<u>F</u> ile	<u>E</u> dit	View	<u>Go</u>	apture	Anal	lyze	<u>S</u> tatis	stics	Tele	phon	<u>у W</u> i	reles	s <u>T</u> ool	s <u>H</u> e	lp						
	5	۲		×	6			∿ [Ð		. 🏼							
(ip.	addr	eq 94.:	140.11	4.6 or	ip.add	lr eq	5.61.3	34.51	.) and	l ssl.h	ands	hake.	type eo	q 11	X	Exp	pression	. + ba	isic basi	c+∣basi	c+dns
Time				Sr	с			port	D	st			port	l	nfo						
20	019-1	1-12	15:0	9 94	4.140).114	4.6	443	1	9.11	.12.	101	4921	5 5	erver	Hello,	Certi	ficate,	Serve	- Hello	Done
20)19-1	.1-12	15:0	9 94	1.140).114	4.6	443	10	9.11	.12.	101	4921	6 5	erver	Hello,	Certi	ficate,	Serve	⁻ Hello	Done
20)19-1	.1-12	15:1	0 94	1.140).114	4.6	443	10	9.11	.12.	101	4921	7 5	erver	Hello,	Certi	ficate,	Serve	- Hello	Done
20)19-1	.1-12	15:1	0 94	1.140	.114	4.6	443	10	9.11	.12.	101	4921	8 5	erver	Hello,	Certi	ficate,	Serve	⁻ Hello	Done
20)19-1	.1-12	15:1	4 5	.61.3	34.53	1	443	10	9.11	.12.	101	4925	0 5	erver	Hello,	Certi	ficate,	Serve	- Hello	Done
20)19-1	.1-12	15:1	4 5	.61.3	34.53	1	443	10	9.11	.12.	101	49282	2 5	erver	Hello,	Certi	ficate,	Serve	⁻ Hello	Done
20)19-1	.1-12	15:1	4 5	.61.3	34.53	1	443	10	9.11	.12.	101	49284	4 5	erver	Hello,	Certi	ficate,	Serve	- Hello	Done
20)19-1	.1-12	15:1	4 5	.61.3	34.53	1	443	10	9.11	.12.	101	4928	7 5	erver	Hello,	Certi	ficate,	Serve	⁻ Hello	Done
4																					►.
Fra	→ Frame 1757: 1155 bytes on wire (9240 bits), 1155 bytes captured (9240 bits)																				
→ Etl	nern	et II	, Sro	: Ne	tgea	r_b6	5:93:	f1	(20	e5:	2a:b	6:93	:f1),	Ds	: Hew	/lettP_1	.c:47:a	e (00:0	98:02:1	c:47:ae)
→ In	tern	et Pr	otoco	ol Ve	rsio	n 4,	Sro	: 9	4.14	10.1	14.6	, Ds	t: 10	9.11	12.10	1					
→ Tra	ansm	issio	n Cor	ntrol	. Pro	toco	bl, S	Src	Port	4	43,	Dst	Port:	492	215, S	seq: 1,	Ack: 1	34, Lei	n: 1101		
- See	cure	Sock	ets L	.ayer		- e	хра	an	ατ	nış		1e									
1	LSV1	L.2 R	ecord	Lay	er: F	Hand	Ishak	e P	roto	COT	: Se	rver	Hell	.0	416				line		
- 1	LSV1	L.2 R	ecord	Lay	er: H	Hand	Ishak	e P	roto	COT	Ce	rtif	icate		e un	en ex	pano	Inis	line		
	Con	tent	туре	: на	nasna	аке	(22)														
	ver	sion	: ILS	1.2	(OXC	0303)														
	Len	gtn:	1001							41				ا ام ه	الملط	line					
· ·	- Han	dsha	ke Pr	otoc	5T: C	cert.	1†1C	ate		• u	ier	\mathbf{e}	cpar	10	Inis I	line					
	н	andsr	аке	rype:	Cer	τ1Τ1	icate	e (1	1)												
	L	engtr	1: 99	<i>.</i>		_															
	С	erti	icate	es Le	ength	1: 99	94	1							_						
	÷ C	ertit	icate	es (s)94 b	ytes	s)		ine	n e	exp	and	a m	S II	ne						
		Cert	11108	ite L	.engt	n: 9	991													17.	
0000	0.0	08	92 1c	47 2	ae 20	e5	2a	h6	93 f	1 08	00	45 6	0		*	F.	(14 0	- 00mm	anNomo-	naltmon	A .
0010	04	75	94 33	00 0	0 80	06	4b	4e	5e 8	c 72	06	0a ()b .	u 3	κ	N^ · r · · ·					
0 2	Ursi	nif-traf	fic-exa	mple-4	1.pcap							54 6			Pa	ckets: 250	9 · Displ	ayed: 8 ((0.3%) F	Profile: De	fault

Figure 22. Working our way to the certificate issuer data in the Dridex traffic.

	Ursnif-traffic-example-4.pcap 🔶 _ 🗆 🗙									
File Edit View	<u>Go</u> <u>Capture</u> <u>Analyze</u> <u>Stat</u>	istics	Telephony <u>W</u> ireles	5 <u>T</u> ools	<u>H</u> elp					
	🗎 🗎 🔀 🗔 🧔 🕏	≁ K								
📕 (ip.addr eq 94	(ip.addr eq 94.140.114.6 or ip.addr eq 5.61.34.51) and ssl.handshake.type eq 11 Expression + basic basic+ »									
Time	Src	port	Dst	port	Info					
2019-11-12	2 15:09 94.140.114.6	443	10.11.12.101	49215	Server Hello	, Certificate,	Server H	ello Done		
2019-11-12	2 15:09 94.140.114.6	443	10.11.12.101	49216	Server Hello	, Certificate,	Server He	ello Done		
2019-11-12	2 15:10 94.140.114.6	443	10.11.12.101	49217	Server Hello	, Certificate,	Server He	ello Done		
2019-11-12	2 15:10 94.140.114.6	443	10.11.12.101	49218	Server Hello	, Certificate,	Server He	ello Done		
2019-11-12	2 15:14 5.61.34.51	443	10.11.12.101	49250	Server Hello	, Certificate,	Server He	ello Done		
2019-11-12	2 15:14 5.61.34.51	443	10.11.12.101	49282	Server Hello	, Certificate,	Server He	ello Done		
2019-11-12	2 15:14 5.61.34.51	443	10.11.12.101	49284	Server Hello	, Certificate,	Server He	ello Done		
2019-11-12	2 15:14 5.61.34.51	443	10.11.12.101	49287	Server Hello	, Certificate,	Server He	ello Done		
•								اط ا		
- Certi	✓ Certificates (994 bytes)									
Cer	tificate Length: 991									
Cer	tificate: 308203db308	3202c3	a0030201020209	00c6b6	794a317884d4	. (id-at-commo	nName=ndl	tman-dsar		
- s:	ignedCertificate									
	version: v3 (2)									
	serialNumber: 143187	654248	335593428							
	signature (sha256Wit	hRSAE	ncryption)					_		
	issuer: rdnSequence	(0)								
	rdnSequence: 5 iter	ns (ic	l-a <u>t-commonNam</u> e	e=ndltr	nan-dsamutb.spi	egel,id-at-or	nanization	alUnitNa		
	RDNSequence item:	1 it	em (id-at-cour	tryNan	ie=NP)	-				
	RDNSequence item:	1 it	em (id-at-loca	lityNa	(me=Kathmandu)					
	RDNSequence item:	1 it	em (id-at-orga	nizati	.onName=Buveców	w Fftaites 0.\	(.E.E.)			
	RDNSequence item:	1 it	em (id-at-orga	nizati	onalUnitName=0	lfo Dusar Lath	na) í	_		
	RDNSequence item:	1 it	em (id-at-com	onName	=ndltman-dsamu	tb.spieael)	,			
•	validity		,			1 3 7				
· ·	subject: rdnSequence	(0)						-		
4					_			•		
0000 00 08	02 1c 47 ae 20 e5 2a	b6 93	3 f1 08 00 <mark>45</mark> 0	00	· · G · · * · · · · · E			^		
0010 04 75	04 33 00 00 80 06 4b	4e 5	e 8c 72 06 0a 0)b ∙u	3 KN∧.r.			-		
😑 🗷 Ursnif-tra	ffic-example-4.pcap				Packets: 25	09 · Displayed: 8 (0	.3%) Profil	e: Default		

Figure 23. Reaching the certificate issuer data from one of the Dridex IP addresses.

Under the *rdnSequence* line, we find properties of the certificate issuer. Certificate issuer characteristics for HTTPS/SSL/TLS traffic at 94.140.114[.]6 follows:

- countryName=NP
- localityName=Kathmandu
- organizationName=Buvecoww Fftaites O.V.E.E.
- organizationalUnitName=Olfo Dusar Latha
- commonName=ndltman-dsamutb.spiegel

Certificate issuer data is different for 5.61.34[.]51, but it follows a similar style:

- countryName=MU
- localityName=Port Louis
- organizationName=Ppoffi Sourinop Cooperative
- organizationalUnitName=ipeepstha and thicioi
- commonName=plledsaprell.Byargt9wailen.voting

This type of issuer data is commonly seen for Dridex post-infection traffic. In our next example, you can further practice reviewing certificate issuer data for Dridex.

Example 5: Evaluation

The fifth pcap for this tutorial, *Ursnif-traffic-example-5.pcap*, is available <u>here</u>. Like our previous example, this pcap has an Ursnif infection followed by Dridex, so we can practice the skills described so far in this tutorial.

Based on what we have learned so far, open the fifth pcap in Wireshark, and answer the following questions:

- For the initial Ursnif binary, which URL returned a Windows executable file?
- After the initial Ursnif binary was sent, the infected Windows host contacted different domains for the HTTP GET requests. Which domain was the traffic successful and allowed the infection to proceed?
- What domain was used in HTTPS traffic after Ursnif became persistent on the infected Windows host?
- What URL ending in .rar was used to send follow-up malware to the infected Windows host?
- What IP addresses were used for the Dridex post-infection traffic?

Answers follow.

Q: For the initial Ursnif binary, which URL returned a Windows executable file?

A: hxxp://ritalislum[.]com/obedle/zarref.php?l=sopopf8.cab

The only Windows executable file in this pcap is the initial Windows executable file for Ursnif. Use the following Wireshark search filter to quickly find this executable:

ip contains "This program"

This filter should provide only one frame in the results. Follow the TCP stream for this frame as shown in Figure 24.

Ursnif-traffic-example-5.pcap 🔶 🗕 🗆 🗙									
<u>F</u> ile <u>E</u> dit <u>V</u> iew <u>G</u> o <u>C</u> apture <u>A</u> r	nalyze <u>S</u> tatistics Telephony <u>W</u>	ireless <u>T</u> ools <u>H</u> elp							
$\blacksquare \blacksquare \boxtimes \otimes \blacksquare \blacksquare \boxtimes \boxtimes \boxtimes \otimes \Rightarrow \Rightarrow \Rightarrow \blacksquare \blacksquare \blacksquare \blacksquare \oplus \bigcirc \bigcirc \bigcirc \blacksquare$									
📕 ip contains "This program"		Expression + basic basic+ basic+dns							
Time Src	port Dst	port Info							
2019-11-21 15:34 178.	57.217.57 80 10.11.23	.101 49158 80 → 49158 [PSH, ACK]							
	<u>M</u> ark/Unmark Packet								
	<u>I</u> gnore/Unignore Packet								
	Set/Unset Time Reference								
	Time Shift								
	Packet Comment								
	Edit Resolved Name								
	Apply as Filter	•							
	Prepare a Filter	•							
4	Conversation Filter								
▶ Frame 68: 1424 bytes o	Colorize Conversation	, bytes capture (1392 bits)							
Ethernet II, Src: Netg	SCTP	93:f1), Dst:							
Internet Protocol Vers Transmission Control D	Follow	TCP Stream							
• Transmission Control P	Copy	Seq: 1, ACK: 96, L							
	Сору	CCL Ctream							
	Protocol Preferences	> SSL Stream							
0000 20 e5 22 b6 93 f1	Decode <u>A</u> s	HIIP Stream							
20 20 20 20 33 11	Show Packet in New <u>W</u> indow	• • • • • • • • • • • • • • • • • • •							
😑 💈 Ursnif-traffic-example-5.pc	ap Packets	7833 · Displayed: 1 (0.0%) Profile: Default							

Figure 24. Filtering to find a frame with the Windows executable file and following the TCP stream.

The TCP stream window contains the domain and URL from the GET request as shown in Figure 25.

Wireshark · Follow TCP Stream (tcp.stream eq 1) · Ursnif-traffic-example-5.pcap							
GET /obedle/zarref.php?l=sopopf8 Host: ritalislum.com Connection: Keep-Alive	B.cab HTTP/1.1						
HTTP/1.1 200 OK Date: Thu, 21 Nov 2019 15:34:51 Server: Apache/2.2.15 (CentOS) X-Powered-By: PHP/7.2.25	URL info for this HTTP request						
Content-Disposition: Attachment;	filename="sopopf8.cab"						
Expires: 0 Cache-Control: must-revalidate Pragma: public Content-Length: 1655808 Connection: close Content-Type: application/octet-stream							
MZ@@	!L.!						
\$	TA.TM.A.20.A.;.@.X.A.\0.A.\:.A.						
1 client pkt, 1,207 server pkts, 1 turn.							
Entire conversation (1,656 kB)	 Show and save data as ASCII Stream 1 						
Find:	Find <u>N</u> ext						
# Help	Filter Out This Stream Print Save as Back X Close						

Figure 25. URL info from the TCP stream.

Q: After the initial Ursnif binary was sent, the infected Windows host contacted different domains for the HTTP GET requests. Which domain was the traffic successful and allowed the infection to proceed?

A: k55gaisi[.]com

Use your *basic* web filter for an overview of the web traffic. HTTP requests caused by this variant of Ursnif start with GET /images/ as already seen in examples two, three, and four of this tutorial. The first HTTP request to k55gaisi[.]com at 15:36 UTC is noted in Figure 26. But if you follow the TCP stream, it shows a 404 Not Found as the response.

	Ursnif-traffic-example-5.pcap 🔷 🕒 📼 🗙									
<u>F</u> ile <u>E</u> dit <u>V</u> iew	<u>Go</u> <u>C</u> apture <u>A</u> nalyze <u>S</u> tatis	stics Te	elephony <u>W</u> ireless <u>T</u> ools <u>H</u> elp							
(http.request	or ssl.handshake.type == 1) ar	Expression + basic basic+ basic+dns								
Time	Dst	port	Host	Info						
	1 15:34 198.70.69.145	80	www.msftncsi.com	GET /ncsi.txt HTTP/1.1						
2019-11-2	1 15:34 178.57.217.57	80	ritalislum.com	GET /obedle/zarref.php?l=sopopf8.cab						
2019-11-2	1 15:35 172.217.12.37	443	gmail.com	Client Hello						
2019-11-2	1 15:35 172.217.12.37	443	gmail.com	Client Hello						
2019-11-2	1 15:35 172.217.12.36	443	www.google.com	Client Hello						
2019-11-2	1 15:35 172.217.12.36	443	www.google.com	Client Hello						
2019-11-2	1 15:36 13.107.21.200	80	www.bing.com	GET /favicon.ico HTTP/1.1						
2019-11-2	1 15:36 72.21.81.200	443	iecvlist.microsoft.com	Client Hello						
2019-11-	1 15:00 70 01 01 000	443	iecvlist.microsoft.com	Client Hello						
2019-11-	404 Not Found I	443	iecvlist.microsoft.com	Client Hello						
2019-11-		443	r20swj13mr.microsoft.com	Client Hello						
2019-11-2	1 15:36 72.21.81.200		r20swj13mr.microsoft.com	Client Hello						
2019-11-2	1 15:36 45.132.19.167	80	k55gaisi.com	GET /images/_2B5RR9mMF8orFi53g/RY2TI						
	Dedirect to	443	iecvlist.microsoft.com	Client Hello						
	Redirectio	443	iecvlist.microsoft.com	Client Hello						
1 110100	arch arrar agm	0	www.bing.com	GET /favicon.ico HTTP/1.1						
*****.50	arch-enor.com	4 80	bon11ljgarry.com	GET /images/GsnEhzNmtNegSXggNDcYx24/N						
2019-11-2	1 15:37 104.124.58.13	5 80	www.search-error.com	GET /search/?q=http%3A//bon11ljgarry.						
2019-11-2	1 15:37 104.124.58.130	5 80	www.search-error.com	GET /s/css/bootstrap.min.css HTTP/1.1						
2019-11-2	1 15:37 104.124.58.130	5 80	www.search-error.com	GET /s/css/searchguide.css HTTP/1.1						
2019-11-2	1 15:37 104.124.58.130	5 80	www.search-error.com	GET /s/css/font-awesome-min.css HTTP/						
4			•							
▶ Frame 39:	151 bytes on wire (120	8 bit	s), 151 bytes captured (1	208 bits)						
→ Ethernet I	I, Src: Netgear_b6:93:	f1 (2	0:e5:2a:b6:93:f1), Dst: N	etgear_b6:93:f1 (20:e5:2a:b6:93:f1)						
→ Internet P	rotocol Version 4, Src	:: 10.	11.21.101, Dst: 198.70.69	. 145						
0000 20 e5	2a b6 93 f1 20 e5 2a	b6 93	f1 08 00 45 00 ·*····	*····E· *						
😑 🕈 Ursnif-tra	affic-example-5.pcap			Packets: 7833 · Displayed: 139 (1.8%) Profile: Default						

Figure 26. Searching web traffic for HTTP GET requests caused by Ursnif.

Also shown in Figure 26, the next HTTP GET request for an Ursnif-style URL is to bon11ljgarry[.]com at 15:37 UTC. The HTTP stream for that request reveals a redirect to a URL at www.search-error[.]com.

Scroll down further, and for similar traffic to leinwqoa[.]com as noted in Figure 27.

	Ursnif-traffic-example-5.pcap	+ _ = ×			
File Edit View Go Capture Analyze Statistics	Telephony <u>W</u> ireless <u>T</u> ools <u>H</u> elp				
	🛊 🏟 📃 🗐 🔍 Q, Q, 🏢				
(http.request or ssl.handshake.type == 1) and !(s	sdp)	Expression + basic basic+ basic+dns			
Time Dst po	rt Host	Info			
2019-11-21 15:37 104.124.58.136 80	www.search-error.com	GET /s/js/searchguide.js HTTP/1.1			
2019-11-21 15:37 104.124.58.136 80	www.search-error.com	GET /s/js/jquery.cookie.js HTTP/1			
2019-11-21 15:37 104.124.58.136 80	www.search-error.com	GET /s/fonts/glyphicons-halflings-			
2019-11-21 15:37 104.124.58.136 80	www.search-error.com	GET /s/img/smob/favicon.ico HTTP/1.1			
2019-11-21 15:37 72.21.81.200 44	3 iecvlist.microsoft.com	Client Hello			
2019-11-21 15:37 72.21.81.200 44	3 iecvlist.microsoft.com	Client Hello			
2019-11-21 15:38 216.58.193.142 80	google.com	GET /images/eeQYmrUT0klCaxJbVow/0MWwc			
2019-11-21 15:38 172.217.12.36 80	www.google.com	GET /images/errors/robot.png HTTP/1.1			
Podiroot to 80	www.google.com	GET /images/branding/googlelogo/1x/go			
	3 iecvlist.microsoft.com	Client Hello			
WWW soarch orror com	iecvlist.microsoft.com	Client Hello			
	leinwqoa.com	GET /images/PdFvuqJOx/bR2kLmJFusrlsh/			
2019-11-21 15:39 104.124.58.136 80	www.search-error.com	GET /search/?q=http%3A//leinwqoa.com/			
2019-11-21 15:39 104.124.58.136 80	www.search-error.com	GET /s/css/bootstrap.min.css HTTP/1.1			
2019-11-21 15:39 104.124.58.136 80	www.search-error.com	GET /s/css/searchguide.css HTTP/1.1			
2019-11-21 15:39 104.124.58.136 80	www.search-error.com	GET /s/css/font-awesome-min.css HTTP/			
2019-11-21 15:39 104.124.58.136 80	www.search-error.com	GET /s/js/jquery.min.js HTTP/1.1			
2019-11-21 15:39 104.124.58.136 80	www.search-error.com	GET /s/js/bootstrap.min.js HTTP/1.1			
2019-11-21 15:39 104.124.58.136 80	www.search-error.com	GET /s/js/searchguide.js HTTP/1.1			
2019-11-21 15:39 104.124.58.136 80	www.search-error.com	GET /s/js/jquery.cookie.js HTTP/1.1			
() Provide the second state (4000 bits) and but second (4000 bits)					
Finally 33: 151 bytes on white (1205 bits), 151 bytes captured (1208 bits)					
Ethernet II, STC: Netgear_Do:93:11 (20:00:93:11), DST: Netgear_DD:93:11 (20:00:23:00:93:11)					
· Internet Protocol Version 4, SIC: 10.11.21.101, DSt: 106.70.00.140					
0000 20 e5 2a b6 93 f1 20 e5 2a b6 93 f1 08 00 45 00 ·*····E·					
		Packets: 7833 · Displayed: 139 (1.8%) Profile: Default			

Figure 27. Finding another Ursnif-style URL that redirects to a search error page.

Scroll down further to find four HTTP GET requests to k55gaisi[.]com that return 200 OK responses. From this point, the Ursnif infection proceeds, and we find no further Ursnif-style HTTP requests that start with GET /images/.

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help Image: Construct Analyze Statistics Telephony Wireless Tools Help Image:				Ursnif-traffic-example-5.pcap	(* <u>-</u> • ×	
Image: Second	File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help					
Image: Interpretendent of the second seco	A = B = B = B = B = B = B = B = B =					
Time Dst port Host Info 2019-11-21 15:39104.124.58.136 80 www.search-error.com GET /s/img/smob/favicon.ico HTTP/1.1 2019-11-21 15:39104.124.58.136 80 www.search-error.com GET /s/fonts/glyphicons-halflings-rec 2019-11-21 15:3972.21.81.200 443 iecvlist.microsoft.com Client Hello 2019-11-21 15:3972.21.81.200 443 iecvlist.microsoft.com Client Hello 2019-11-21 15:39	(http.request o	or ssl.handshake.type == 1) and	d !(ssd	p)	Expression + basic basic+ basic+dns	
2019-11-21 15:39104.124.58.136 80 www.search-error.com GET /s/img/smob/favicon.ico HTTP/1.1 2019-11-21 15:39104.124.58.136 80 www.search-error.com GET /s/fonts/glyphicons-halflings-rec 2019-11-21 15:39104.124.58.136 80 www.search-error.com GET /s/fonts/glyphicons-halflings-rec 2019-11-21 15:3972.21.81.200 443 iecvlist.microsoft.com Client Hello 2019-11-21 15:3972.217.12.37 443 gmail.com Client Hello 2019-11-21 15:39	Time	Dst	port	Host	Info	
2019-11-21 15:39 104.124.58.136 80 www.search-error.com GET /s/fonts/glyphicons-halflings-rec 2019-11-21 15:39 72.21.81.200 443 iecvlist.microsoft.com Client Hello 2019-11-21 15:39 72.21.81.200 443 iecvlist.microsoft.com Client Hello 2019-11-21 15:39 72.21.81.200 443 iecvlist.microsoft.com Client Hello 2019-11-21 15:39 172.217.12.37 443 gmail.com Client Hello 2019-11-21 15:40 72.21.81.200 443 www.google.com Client Hello 2019-11-21 15:40 72.21.81.200 443 iecvlist.microsoft.com Client Hello 2019-11-21 15:40 72.21.81.200 443 iecvlist.microsoft.com Client Hello 2019-11-21 15:40 72.21.81.200 443 iecvlist.microsoft.com Client Hello 2019-11-21 15:40 72.21.81	2019-11-21	L 15:39 104.124.58.136	80	www.search-error.com	GET /s/img/smob/favicon.ico HTTP/1.1	
2019-11-21 15:39 104.124.58.136 80 www.search-error.com GET /s/fonts/glyphicons-halflings-rec 2019-11-21 15:39 72.21.81.200 443 iecvlist.microsoft.com Client Hello 2019-11-21 15:39 72.21.81.200 443 gmail.com Client Hello 2019-11-21 15:39 172.217.12.37 443 gmail.com Client Hello 2019-11-21 15:40 72.21.81.200 443 www.google.com Client Hello 2019-11-21 15:40 72.21.81.200 443 iecvlist.microsoft.com Client Hello 2019-11-21 15:40 45.132.19.167 80 K55gaisi.com GET /images/W226kfzMbrkgMuVj7zer/Dchi 2019-11-21 15:40 45.132.19.167 80 K55gaisi.com GET /images/KZ5haidgeKCTVKI/U0Q6ZV 2019-11-21 15:40 45.132.19.167 80 K55gaisi.com Client Hello 2019-11-21 15:40 45.8	2019-11-21	L 15:39 104.124.58.136	80	www.search-error.com	GET /s/fonts/glyphicons-halflings-reo	
2019-11-21 15:3972.21.81.200 443 iecvlist.microsoft.com Client Hello 2019-11-21 15:3972.21.81.200 443 iecvlist.microsoft.com Client Hello 2019-11-21 15:3972.217.12.37 443 gmail.com Client Hello 2019-11-21 15:3972.21.81.200 443 www.google.com Client Hello 2019-11-21 15:4045.132.19.167 80 k55gaisi.com GET /fmages/We26kfzMbrKgMuVj7zer/Dch2 2019-11-21 15:4045.132.19.167 80 k55gaisi.com GET /fmages/dLK6lmidgTekCTVKI/U0QGzV 2019-11-21 15:4045.132.19.167 80 k55gaisi.com GET /fmages/dLK6lmidgTekCTVKI/U0QGzV 2019-11-21 15:4072.21.81.200 443 iecvlist.microsoft.com Client Hello 2019-11-21 15:4072.21.81.200 443 iecvlist.microsoft.com Client Hello 2019-11-21 15:41172.217.12.37 443 gmail.com Client Hello 2019-11-21 15:41172.217.12.37 443 google.com <td>2019-11-21</td> <td>L 15:39 104.124.58.136</td> <td>80</td> <td>www.search-error.com</td> <td>GET /s/fonts/glyphicons-halflings-rec</td>	2019-11-21	L 15:39 104.124.58.136	80	www.search-error.com	GET /s/fonts/glyphicons-halflings-rec	
2019-11-21 15:3972.21.81.200 443 iecvlist.microsoft.com Client Hello 2019-11-21 15:39172.217.12.37 443 gmail.com Client Hello 2019-11-21 15:39172.217.12.37 443 gmail.com Client Hello 2019-11-21 15:39172.217.12.37 443 gmail.com Client Hello 2019-11-21 15:39200.0K 443 www.google.com Client Hello 2019-11-21 15:4072.21.81.200 443 www.google.com Client Hello 2019-11-21 15:4072.21.81.200 443 iecvlist.microsoft.com Client Hello 2019-11-21 15:4045.132.19.167 80 k55gaisi.com GET /images/We26kfzMbrKgMuVj7zer/Dch: 2019-11-21 15:4045.132.19.167 80 k55gaisi.com GET /images/dDLX6lmidgTeKCTVKI/U0Qczv 2019-11-21 15:4045.132.19.167 80 k55gaisi.com Client Hello 2019-11-21 15:4072.21.81.200 443 iecvlist.microsoft.com Client Hello 2019-11-21 15:41172.217.12.37 443 google.com Client Hello 2019-11-21 2019-11-21 15:41172.217.12.3	2019-11-21	L 15:39 72.21.81.200	443	iecvlist.microsoft.com	Client Hello	
2019-11-21 15:39 172.217.12.37 443 gmail.com Client Hello 2019-11-21 15:39 172.217.12.37 443 gmail.com Client Hello 2019-11-21 15:39 200 OK 443 www.google.com Client Hello 2019-11-21 15:39 200 OK 443 www.google.com Client Hello 2019-11-21 15:40 72.21.81.200 443 iecvlist.microsoft.com Client Hello 2019-11-21 15:40 45.132.19.167 80 k55gaisi.com GET /fmages/We26kfzMbrKgMuVj7zer/Dch2 2019-11-21 15:40 45.132.19.167 80 k55gaisi.com GET /fmages/Mc26kfzMbrKgMuVj7zer/Dch2 2019-11-21 15:40 45.132.19.167 80 k55gaisi.com GET /images/Mc26kfzMbrKgMuVj7zer/Dch2 2019-11-21 15:40 72.21.81.200 443 iecvlist.microsoft.com Client Hello 2019-11-21 15:40 72.21.81.200 443 iecvlist.microsoft.com Client Hello 2019-11-21 15:41 172.21.7.12.37 443 google.com Client Hello 2019-11-21 istat	2019-11-21	L 15:39… 72.21.81.200	443	iecvlist.microsoft.com	Client Hello	
2019-11-21 15:39 172.217.12.37 443 gmail.com Client Hello 2019-11-21 15:39 200 OK 443 www.google.com Client Hello 2019-11-21 15:39 200 OK 443 www.google.com Client Hello 2019-11-21 15:40 72.21.81.200 443 iecvlist.microsoft.com Client Hello 2019-11-21 15:40 45.132.19.167 80 k55gaisi.com GET /images/We26kfzMbrKgMuVj7zer/Dchz 2019-11-21 15:40 45.132.19.167 80 k55gaisi.com GET /images/We26kfzMbrKgMuVj7zer/Dchz 2019-11-21 15:40 45.132.19.167 80 k55gaisi.com GET /images/MDLX6LmidgTeKCTVKI/U0QGzV 2019-11-21 15:40 45.132.19.167 80 k55gaisi.com GET /images/KZblmidgTeKCTVKI/U0QGzV 2019-11-21 15:40 72.21.81.200 443 lecvlist.microsoft.com Client Hello 2019-11-21 15:40 72.21.81.200 443 lecvlist.microsoft.com Client Hello 2019-11-21 15:41 16:58.133.142 443 gogle.com Client Hello 2019-11-21	2019-11-21	L 15:39 172.217.12.37	443	gmail.com	Client Hello	
2019-11-21 15:39. 200 OK 443 www.google.com Client Hello 2019-11-21 15:40. 72.21.81.200 443 iecvlist.microsoft.com Client Hello 2019-11-21 15:40. 72.21.81.200 443 iecvlist.microsoft.com Client Hello 2019-11-21 15:40 45.132.19.167 80 K55gaisi.com GET /images/We26kfzMbrKgMuVj7zer/Dchz 2019-11-21 15:40 45.132.19.167 80 K55gaisi.com GET /images/dDLX6lmidgTeKCTVKI/UoQ6zv 2019-11-21 15:40 45.132.19.167 80 K55gaisi.com GET /images/Ktzhfa0qx_2FOP9lbo/Hmw5X 2019-11-21 15:40 45.132.19.167 80 K55gaisi.com Client Hello 2019-11-21 15:40 72.21.81.200 443 iecvlist.microsoft.com Client Hello 2019-11-21 15:40 72.21.81.200 443 iecvlist.microsoft.com Client Hello 2019-11-21 15:41 172.21.71.2.37 443 google.com Client Hello 2019-11-21 15:41 172.21.81.200 443 iecvlist.microsoft.com Client Hello 2019-11-2	2019-11-21	L 15:39 172.217.12.37	443	gmail.com	Client Hello	
2019-11-21 15:39 200 OK 443 www.google.com Client Hello 2019-11-21 15:40 72.21.81.200 443 iecvlist.microsoft.com Client Hello 2019-11-21 15:40 45.132.19.167 80 k55gaisi.com GET /images/We26kfzMbrKgMuVj7zer/Dch; 2019-11-21 15:40 45.132.19.167 80 k55gaisi.com GET /images/dDLX6lmidgTekCTVKI/UoQGzV 2019-11-21 15:40 45.132.19.167 80 k55gaisi.com GET /images/dDLX6lmidgTekCTVKI/UoQGzV 2019-11-21 15:40 45.132.19.167 80 k55gaisi.com GET /images/KTzhfa0qX_2F0P9lbo/Hmw5X) 2019-11-21 15:40 72.21.81.200 443 iecvlist.microsoft.com Client Hello 2019-11-21 15:40 72.21.81.200 443 iecvlist.microsoft.com Client Hello 2019-11-21 15:41 172.21.71.22.37 443 google.com Client Hello 2019-11-21 15:41 172.21.81.200 443 iecvlist.microsoft.com Client Hello 2019-11-21 15:41 172.21.81.200 443 iecvlist.microsoft.com Client Hello	2019-11-21	L 15:39	443	www.google.com	Client Hello	
2019-11-21 15:4072.21.01.200 443 iecvlist.microsoft.com Client Hello 2019-11-21 15:4072.21.81.200 10 iecvlist.microsoft.com Client Hello 2019-11-21 15:4045.132.19.167 80 k55gaisi.com GET /images/We26kfzMbrKgMuVj7zer/Dchz 2019-11-21 15:4045.132.19.167 80 k55gaisi.com GET /images/dDLX6lmidgTeKCTVKI/UoQGzV 2019-11-21 15:4045.132.19.167 80 k55gaisi.com GET /images/KTzhfa0qX_2FOP91bo/Hmw5XX 2019-11-21 15:4045.132.19.167 80 k55gaisi.com Client Hello 2019-11-21 15:4045.132.19.167 80 k55gaisi.com GET /images/KTzhfa0qX_2FOP91bo/Hmw5XX 2019-11-21 15:4072.21.81.200 443 iecvlist.microsoft.com Client Hello 2019-11-21 15:41172.217.12.37 443 gogle.com Client Hello 2019-11-21 15:41185.118.165.1443 n9maryjanef.com Client Hello Pient Hello 2019-11-21 15:41185.118.165.1443 n9maryjanef.com Client Hello Pient Hello Pient Hello 2019-11-21 15:41185.118.165.1443 n9maryjanef.com	2019-11-21	15:39 200 OK	443	www.google.com	Client Hello	
2019-11-21 15:40 72.21.81.200 1 iecvlist.microsoft.com Client Hello 2019-11-21 15:40 45.132.19.167 80 k55gaisi.com GET /images/We26kfzMbrKgMuVj7zer/Dchz 2019-11-21 15:40 45.132.19.167 80 k55gaisi.com GET /favicon.ico HTTP/1.1 2019-11-21 15:40 45.132.19.167 80 k55gaisi.com GET /images/dDLX6lmidgTeKCTVKI/U0QGzv 2019-11-21 15:40 45.132.19.167 80 k55gaisi.com GET /images/KTzhfaOqX_2FOP9lbo/Hmw5X 2019-11-21 15:40 72.21.81.200 443 iecvlist.microsoft.com Client Hello 2019-11-21 15:40 72.21.81.200 443 iecvlist.microsoft.com Client Hello 2019-11-21 15:41 172.21.81.200 443 iecvlist.microsoft.com Client Hello 2019-11-21 15:41 172.21.81.200 443 iecvlist.microsoft.com Client Hello 2019-11-21 15:41 172.21.81.200 443 iecvlist.microsoft.com Client Hello 2019-11-21 15:41 165.8.193.142 443 google.com Client Hello	2019-11-21	L 15:40 72.21.01.200	443	iecvlist.microsoft.com	Client Hello	
2019-11-21 15:40 45.132.19.167 80 k55gaisi.com GET /images/We26kfzMbrKgMuVj7zer/Dchz 2019-11-21 15:40 45.132.19.167 80 k55gaisi.com GET /favicon.ico HTTP/1.1 2019-11-21 15:40 45.132.19.167 80 k55gaisi.com GET /images/dDLX6ImidgTeKCTVKI/U0QGzV 2019-11-21 15:40 45.132.19.167 80 k55gaisi.com GET /images/dDLX6ImidgTeKCTVKI/U0QGzV 2019-11-21 15:40 72.21.81.200 443 lecvlist.microsoft.com Client Hello 2019-11-21 15:41 72.21.81.200 443 lecvlist.microsoft.com Client Hello 2019-11-21 15:41 72.21.81.200 443 lecvlist.microsoft.com Client Hello 2019-11-21 15:41 172.217.12.37 443 google.com Client Hello 2019-11-21 15:41 185.118.165.1 443 n9maryjanef.com Client Hello 2019-11-21 15:41 185.118.165.1 443 n9maryjanef.com Client Hello 2019-11-21 15:41 185.118.165.1 443 n9maryjanef.com Client Hello	2019-11-21	L 15:40 72.21.81.200	4	<pre>iecvlist.microsoft.com</pre>	Client Hello	
2019-11-21 15:40 45.132.19.167 80 k55gaisi.com GET /favicon.ico HTTP/1.1 2019-11-21 15:40 45.132.19.167 80 k55gaisi.com GET /images/dDLX6lmidgTeKCTVKI/U0QGzv 2019-11-21 15:40 45.132.19.167 80 k55gaisi.com GET /images/dDLX6lmidgTeKCTVKI/U0QGzv 2019-11-21 15:40 45.132.19.167 80 k55gaisi.com GET /images/dDLX6lmidgTeKCTVKI/U0QGzv 2019-11-21 15:40 72.21.81.200 443 iecvlist.microsoft.com Client Hello 2019-11-21 15:40 72.21.81.200 443 iecvlist.microsoft.com Client Hello 2019-11-21 15:41 172.21.71.2.37 443 gmail.com Client Hello 2019-11-21 15:41 126.58.193.142 443 google.com Client Hello 2019-11-21 15:41 185.118.165.1 443 n9maryjanef.com Client Hello 2019-11-21 15:41 185.118.165.1 443 n9maryjanef.com Client Hello 2019-11-21 15:41 185.118.165.1 443 n9maryjanef.com Client Hello	2019-11-21	L 15:40 45.132.19.167	80	k55gaisi.com	GET /images/We26kfzMbrKgMuVj7zer/Dchz	
2019-11-21 15:40 45.132.19.167 80 k55gaisi.com GET /images/dDLX6lmidgTeKCTVKI/UoQGzv 2019-11-21 15:40 45.132.19.167 80 k55gaisi.com GET /images/dDLX6lmidgTeKCTVKI/UoQGzv 2019-11-21 15:40 72.21.81.200 443 lecvlist.microsoft.com Client Hello 2019-11-21 15:40 72.21.81.200 443 lecvlist.microsoft.com Client Hello 2019-11-21 15:41 172.217.12.37 443 gmail.com Client Hello 2019-11-21 15:41 172.217.12.37 443 google.com Client Hello 2019-11-21 15:41 185.118.165.1 443 google.com Client Hello 2019-11-21 15:41 185.118.165.1 443 n9maryjanef.com Client Hello 1 Frame 39:	2019-11-21	L 15:40 45.132.19.167	80	k55gaisi.com	GET /favicon.ico HTTP/1.1	
2019-11-21 15:40 45.132.19.167 80 k55gaisi.com GET /images/KTzhfa0qX_2F0P9lbo/Hmw5X) 2019-11-21 15:40 72.21.81.200 443 lecvlist.microsoft.com Client Hello 2019-11-21 15:40 72.21.81.200 443 lecvlist.microsoft.com Client Hello 2019-11-21 15:41 72.21.81.200 443 iecvlist.microsoft.com Client Hello 2019-11-21 15:41 172.217.12.37 443 gmail.com Client Hello 2019-11-21 15:41 172.217.12.37 443 google.com Client Hello 2019-11-21 15:41 165.8.193.142 443 google.com Client Hello 2019-11-21 15:41 185.118.165.1 443 n9maryjanef.com Client Hello 1 Frame 39: 151 <t< td=""><td>2019-11-21</td><td>L 15:40 45.132.19.167</td><td>80</td><td>k55gaisi.com</td><td>GET /images/dDLX6lmidgTeKCTVKI/UoQGzV</td></t<>	2019-11-21	L 15:40 45.132.19.167	80	k55gaisi.com	GET /images/dDLX6lmidgTeKCTVKI/UoQGzV	
2019-11-21 15:40 72.21.81.200 443 lecvlist.microsoft.com Client Hello 2019-11-21 15:40 72.21.81.200 443 iecvlist.microsoft.com Client Hello 2019-11-21 15:41 172.21.81.200 443 iecvlist.microsoft.com Client Hello 2019-11-21 15:41 172.21.7.12.37 443 gmail.com Client Hello 2019-11-21 15:41 172.217.12.37 443 google.com Client Hello 2019-11-21 15:41 126.58.193.142 443 google.com Client Hello 2019-11-21 15:41 185.118.165.1 443 n9maryjanef.com Client Hello * Frame 39:151 bytes on wire (1208 bits), 151 bytes captured (1208 bits) * * Ethernet II, Src: Netgear_b6:93:f1 <td>2019-11-21</td> <td>L 15:40 45.132.19.167</td> <td>80</td> <td>k55gaisi.com</td> <td>GET /images/KTzhfaOqX_2FOP9lbo/Hmw5X)</td>	2019-11-21	L 15:40 45.132.19.167	80	k55gaisi.com	GET /images/KTzhfaOqX_2FOP9lbo/Hmw5X)	
2019-11-21 15:40 72.21.81.200 443 iecvlist.microsoft.com Client Hello 2019-11-21 15:41 172.217.12.37 443 gmail.com Client Hello 2019-11-21 15:41 172.217.12.37 443 gmail.com Client Hello 2019-11-21 15:41 172.217.12.37 443 gmail.com Client Hello 2019-11-21 15:41 16.58.193.142 443 google.com Client Hello 2019-11-21 15:41 185.118.165.1 443 n9maryjanef.com Client Hello 2019-11-21 15:41 185.118.165.1 443 n9maryjanef.com Client Hello * Frame 39: 151 bytes on wire (1208 bits), 151 bytes captured (1208 bits) * * Ethernet II, Src: Netgear_b6:93:f1 (20:e5:2a:b6:93:f1), Dst: Netgear_b6:93:f1 (20:e5:2a:b6:93:f1) * * Internet Protocol Version 4, Src: 10.11.21.101, Dst: 198.70.69.145 * * 0000 20 e5 2a b6 93 f1 20 e5 2a b6 93 f1 08 00 45 00 * * Urspif-traffic-example-5.pcap * * Packets: 7833 · Displayed: 139 (1.8%) Profile: Default	2019-11-21	L 15:4072.21.81.200	443	iecvlist.microsoft.com	Client Hello	
2019-11-21 15:41 172.217.12.37 443 gmail.com Client Hello 2019-11-21 15:41 26.58.193.142 443 google.com Client Hello 2019-11-21 15:41 26.58.193.142 443 google.com Client Hello 2019-11-21 15:41 185.118.165.1 443 n9maryjanef.com Client Hello Image: state sta	2019-11-21	l 15:40… 72.21.81.200	443	iecvlist.microsoft.com	Client Hello	
2019-11-21 15:41 216.58.193.142 443 google.com Client Hello 2019-11-21 15:41 185.118.165.1 443 n9maryjanef.com Client Hello * Frame 39: 151 bytes on wire (1208 bits), 151 bytes captured (1208 bits) * Ethernet II, Src: Netgear_b6:93:f1 (20:e5:2a:b6:93:f1), Dst: Netgear_b6:93:f1 (20:e5:2a:b6:93:f1) * Internet Protocol Version 4, Src: 10.11.21.101, Dst: 198.70.69.145 ************************************	2019-11-21	L 15:41 172.217.12.37	443	gmail.com	Client Hello	
2019-11-21 15:41 185.118.165.1 443 n9maryjanef.com Client Hello Image: Comparison of the system of the syste	2019-11-21	L 15:41 216.58.193.142	443	google.com	Client Hello	
<pre> Frame 39: 151 bytes on wire (1208 bits), 151 bytes captured (1208 bits) Frame 39: 151 bytes on wire (1208 bits), 151 bytes captured (1208 bits) Ethernet II, Src: Netgear_b6:93:f1 (20:e5:2a:b6:93:f1), Dst: Netgear_b6:93:f1 (20:e5:2a:b6:93:f1) Internet Protocol Version 4, Src: 10.11.21.101, Dst: 198.70.69.145 0000 20 e5 2a b6 93 f1 20 e5 2a b6 93 f1 08 00 45 00 ******************************</pre>	2019-11-21	L 15:41 185.118.165.1	443	n9maryjanef.com	Client Hello	
Frame 39: 151 bytes on wire (1208 bits), 151 bytes captured (1208 bits) Frame 39: 151 bytes on wire (1208 bits), 151 bytes captured (1208 bits) Ethernet II, Src: Netgear_b6:93:f1 (20:e5:2a:b6:93:f1), Dst: Netgear_b6:93:f1 (20:e5:2a:b6:93:f1) Internet Protocol Version 4, Src: 10.11.21.101, Dst: 198.70.69.145 0000 20 e5 2a b6 93 f1 20 e5 2a b6 93 f1 08 00 45 00 ** ** ** ** ** ** Packets: 7833 · Displayed: 139 (1.8%) Profile: Default						
<pre>> Ethernet II, Src: Netgear_06:93:f1 (20:e5:2a:b6:93:f1), Dst: Netgear_b6:93:f1 (20:e5:2a:b6:93:f1) > Internet Protocol Version 4, Src: 10.11.21.101, Dst: 198.70.69.145 0000 20 e5 2a b6 93 f1 20 e5 2a b6 93 f1 08 00 45 00 ** ** ** ** E* 0 20 e5 2a b6 93 f1 20 e5 2a b6 93 f1 08 00 45 00 ** ** ** ** ** ** ** ** ** ** ** **</pre>	Frame 39: 151 bytes on wire (1208 bits), 151 bytes captured (1208 bits)					
Internet Protocol Version 4, Src: 10.11.21.101, Dst: 198.70.69.145 20 e5 2a b6 93 f1 20 e5 2a b6 93 f1 08 00 45 00 *******************************	<pre>> Ethernet II, Src: Netgear_b6:93:f1 (20:e5:2a:b6:93:f1), Dst: Netgear_b6:93:f1 (20:e5:2a:b6:93:f1)</pre>					
0000 20 e5 2a b6 93 f1 08 00 45 00 ************************************	> Internet Protocol version 4, Src: 10.11.21.101, Dst: 198.70.69.145					
O ℤ Ursnif-traffic-example-5.pcap Packets: 7833 · Displayed: 139 (1.8%) Profile: Default	0000 20 e5 2a b6 93 f1 20 e5 2a b6 93 f1 08 00 45 00 ·*····E·					
	😑 💈 Ursnif-tra	ffic-example-5.pcap			Packets: 7833 · Displayed: 139 (1.8%) Profile: Default	

Figure 28. Finding the Ursnif-style HTTP GET requests that return a 200 OK.

Q: What domain was used in HTTPS traffic after Ursnif became persistent on the infected Windows host?

A: n9maryjanef[.]com

When Ursnif is persistent, we no longer see Ursnif-style HTTP requests starting with GET /images/. Instead, we find Ursnif-related HTTPS traffic. Shortly after the final Ursnif-style HTTP GET request, HTTPS traffic to n9maryjanef[.]com begins on 185.118.165[.]109 as highlighted in Figure 29. This is Ursnif traffic.

Ursnif-traffic-example-5.pcap 🔶 🗉 🗙							
<u>File Edit V</u> iew	<u>Go</u> <u>Capture</u> <u>Analyze</u> <u>Statistic</u>	cs Telephon <u>y W</u> ireless <u>T</u> ools <u>H</u> elp					
	A B B						
(http.request o	or ssl.handshake.type == 1) and	!(ssdp) ⊠ →	Expression + basic basic+ basic+dns				
Time	Dst	port Host	Info				
2019-11-21	15:4072.21.81.200	443 iecvlist.microsoft.com	Client Hello				
2019-11-21	. 15:40 72.21.81.200	<pre>443 iecvlist.microsoft.com</pre>	Client Hello				
2019-11-21	. 15:40 45.132.19.167	80 k55gaisi.com	GET /images/We26kfzMbrKgMu				
2019-11-21	. 15:40 45.132.19.167	80 k55gaisi.com	GET /favicon.ico HTTP/1.1				
2019-11-21	. 15:40 45.132.19.167	80 k55gaisi.com	GET /images/dDLX6lmidgTeKC1.				
2019-11-21	15:40 45.132.19.167	80 k55gaisi.com	GET /images/KTzhfaOqX_2F0P91bd				
2019-11-21	15:4072.21.81.200	<pre>443 iecvlist.microsoft.com</pre>	Client Hello				
2019-11-21	15:4072.21.81.200	<pre>443 iecvlist.microsoft.com</pre>	Client Hello				
2019-11-21	. 15:41 172.217.12.37	443 gmail.com	Client Hello				
2019-11-21	15:41 216.58.193.142	443 google.com	Client Hello				
2019-11-21	. 15:41 <mark>185.118.165.10</mark> 9	443 n9maryjanef.com	Client Hello				
2019-11-21	. 15:41 72.21.81.240	80 www.download.windowsupdate.com	n GET /msdownload/update/v3/stat				
2019-11-21	. 15:46 <mark>185.118.165.10</mark> 9	443 n9maryjanef.com	Client Hello				
2019-11-21	. 15:51 185.118.165.109	443 n9maryjanef.com	Client Hello				
2019-11-21	. 15:51 185.118.165.109	443 n9maryjanef.com	Client Hello				
+ Frame 39: 151 bytes on Wire (1208 bits), 151 bytes captured (1208 bits)							
→ Internet Protocol Version 4, Src: 10.11.21.101, Dst: 198.70.69.145							
0000 20 e5 2a b6 93 f1 20 e5 2a b6 93 f1 08 00 45 00 ·*····E·							
O ℤ Ursnif-traffic-example-5.pcap Packets: 7833 · Displayed: 139 (1.8%) Profile: Default							

Figure 29. HTTPS traffic caused by Ursnif.

You can confirm this is Ursnif traffic by filtering on *ip.addr eq 185.118.165.109 and ssl.handshake.type == 11* and reviewing the certificate issuer data. The certificate issuer data should look the same as our second example in Figure 10.

Q: What URL ending in .rar was used to send follow-up malware to the infected Windows host?

A: hxxps://testedsolutionbe[.]com/wp-content/plugins/apikey/uaasdqweeeeqsd.rar

HTTP GET requests caused by Ursnif for follow-up malware end in .rar, so use the following filter to find this URL in our pcap:

http.request and ip contains .rar

The results should be similar to what we see in Figure 30.



Figure 30. Finding the URL for follow-up malware from this Ursnif infection.

Notice in Figure 30 how the HTTP GET request in Figure 30 redirects to an HTTPS URL.

Q: What IP addresses were used for the Dridex post-infection traffic?

A: 185.99.133[.]38 and 5.61.34[.]51

One of these IP addresses is the same as Dridex in our fourth pcap, and it has the same certificate issuer data. Dridex traffic to 185.99.133[.]38 has the same style of certificate issuer data as seen in example 4. Traffic to both IP addresses does not involve a domain name.

The Dridex post-infection traffic is easy to spot in this example if we look for any HTTPS/SSL/TLS traffic without a domain after the HTTP GET request ending in .rar as shown in Figure 31.

		Ursnif-traffic-example-5.pcap			+ - • ×
<u>File Edit View Go</u> Capture Ana	alyze <u>S</u> tatistics Te	lephon <u>y W</u> ireless <u>T</u> ools <u>H</u> elp			
	🗟 💠 🔶 🛠 除	斜 📃 🖲 Q Q 🛙 🎹			
(http.request or ssl.handshake.ty	pe == 1) and !(ssdp)	Express	ion + basic basic+	basic+dns
Time Dst	port	Host	Info		^
2019-11-21 15:51 104.3	1.89.212 80	testedsolutionbe.com	GET /wp-content	/plugins/apikey/uaa	sdqwe
2019-11-21 15:51 <u>104.3</u>	1.89.212 443	testedsolutionbe.com	Client Hello		
2019-11-21 15:51 185.9	9.133.38 443		Client Hello		
2019-11-21 15:51 185.9	9.133.38 443	T	Client Hello	LIRI	
2019-11-21 15:51 185.9	9.133.38 443	I raπic caused	Client Hello		
2019-11-21 15:52 185.9	9.133.38 443	In the Daristan	Client Hello	in rar	
2019-11-21 15:55 5.61.	34.51 443	by Dridex	Client Hello		
2019-11-21 15:55 5.61.	34.51 443	•	Client Hello		
2019-11-21 15:555.61.	34.51 443		Client Hello		
2019-11-21 15:56 185.1	18.165.109 443	n9maryjanef.com	Client Hello		
2019-11-21 15:56 185.1	18.165.109 443	n9maryjanef.com	Client Hello		
2019-11-21 16:01 185.1	18.165.109 443	n9maryjanef.com	Client Hello		
2019-11-21 16:01 185.1	18.165.109 443	n9maryjanef.com	Client Hello		
2019-11-21 16:06 185.1	18.165.109 443	n9maryjanef.com	Client Hello 🏅		
2019-11-21 16:07 172.2	17.12.37 443	gmail.com	Client Hello	∟HTTPS traffi	C I
2019-11-21 16:07 216.5	8.193.142 443	google.com	Client Hello		~
2019-11-21 16:07 185.1	18.165.109 443	n9maryjanef.com	Client Hello	caused by U	rsnif
2019-11-21 16:08 185.1	18.165.109 443	n9maryjanef.com	Client Hello 🎽		
2019-11-21 16:11 185.1	18.165.109 443	n9maryjanef.com	Client Hello 🦰		
2019-11-21 16:11 185.1	18.165.109 443	n9maryjanef.com	Client Hello		
Frame 39: 151 bytes on V	wire (1208 bits	b), 151 bytes captured	(1208 DICS)	E1 (00.05.00.00.00.0	-
> Ethernet II, Src: Netgea	ar_06:93:T1 (20	D:e5:2a:D6:93:T1), DST:	Netgear_b6:93:1	(20:e5:2a:b6:93:T	I)
> Internet Protocol Versio	5H 4, Src: 10.1	LI.21.101, DST: 198.70.	09.145		*
0000 20 e5 2a b6 93 f1 20	0 e5 2a b6 93	f1 08 00 45 00 ·*···	· *····E·		A
⊖ ℤ Ursnif-traffic-example-5.pca	p		Packets: 7833 · Dis	played: 139 (1.8%) Profil	e: Default

Figure 31. Finding the Dridex traffic in our fifth pcap.

Conclusion

This tutorial provided tips for examining Windows infections with Ursnif malware. More pcaps with examples of Ursnif activity can be found at <u>malware-traffic-analysis.net</u>.

For more help with Wireshark, see our previous tutorials:

Get updates from Palo Alto Networks!

Sign up to receive the latest news, cyber threat intelligence and research from us

By submitting this form, you agree to our <u>Terms of Use</u> and acknowledge our <u>Privacy</u> <u>Statement</u>.