VB2021
localhost

7 - 8 October, 2021 / vblocalhost.com

MULTI-UNIVERSE OF ADVERSARY: MULTIPLE CAMPAIGNS OF LAZARUS GROUP AND ITS CONNECTION

Seongsu Park

Kaspersky, Republic of Korea

seongsup4rk@gmail.com

ABSTRACT

The cyber threat landscape is one that changes at an incredibly rapid pace, and the characteristics of various threat actors often evolve at the same pace. Although some threat actors maintain their strategies for a long time, sophisticated actors continuously sharpen their tools, techniques and procedures (TTPs) and aggressively adopt new attack methodologies. Perhaps no other threat actor exhibits this trend more than the Lazarus group: an extremely prolific threat actor operating globally and linked to such high-profile and devastating cyber attacks as the *Sony Pictures Entertainment* hack, several cryptocurrency hacks, and the WannaCry worm outbreak.

Unlike other state-sponsored threat actors, Lazarus uses its attack campaigns to achieve a variety of aims, from cyber espionage to financial profit. In fact, it is because of this broad spectrum of goals that the group keeps expanding the scale of its cyber attacks and drawing the attention of the security community. Untangling and connecting the various waves of the group's activity has become no small undertaking, and different security researchers have their own methodologies and conclusions. In this paper we will focus on a relatively novel malware cluster of the Lazarus group, as well as our perspective on the various representative clusters of the Lazarus group, their connections with one another, and their differences.

INTRODUCTION

After the *Sony Pictures Entertainment* hack and the publication of the Operation BlockBuster [1] research, the Lazarus group began to garner much more attention from the security industry. When Lazarus first began operating, they used a small malware cluster and few cyber attack capabilities. We have continued to track this malware cluster, named Manuscrypt. In its early stages, the Manuscrypt cluster utilized simple malware sets and was used for an extended period to achieve various goals.

However, Manuscrypt's modus operandi began to change in 2018. Several malware clusters started to spin-off from the original malware and were developed independently, creating new malware strains and attack methodologies. Today, Lazarus has become one of the biggest and most notable threat actors with sufficient capabilities to attack various industries simultaneously. As the scale of the Lazarus group increases, many security vendors have published their perspectives on attribution and clustering. Such endeavours by security vendors are always positive in nature, as they're a vital part of keeping the community informed about threat actors' progress and latest activity. However, such efforts can occasionally lead to confusion because of the difference in standards. The attribution of the Lazarus group is incredibly tangled these days, but, given the group's highly prolific nature, it's a situation that needs clarity. That is why we would like to share our perspective on this group and its representative clusters, starting with Manuscrypt.

Figure 1 represents a simple timeline of the Lazarus group's clusters. It contains a relatively new cluster after 2018. This timeline is based on the time when the malware was first discovered or its compilation time. Most clusters originate from the Manuscrypt cluster. Although the Lazarus group used the Manuscrypt variant until recently, its usage rate decreased dramatically as malware clusters and attack methodologies began to diversify.

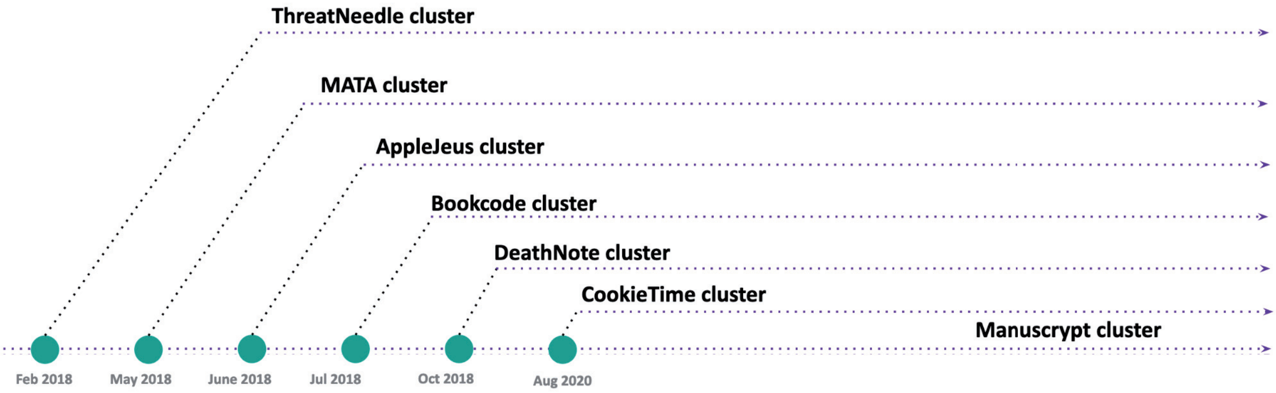


Figure 1: Lazarus malware history.

INTRODUCTION OF CLUSTERS

The clusters mentioned in this report are not all of the clusters utilized by the Lazarus group. Rather, we describe the most commonly used and well-known clusters to us. We have published information about several clusters publicly, however, some are unfamiliar to the public so we will add a brief introduction to each of them.

AppleJeus

Kaspersky published details of Operation AppleJeus and its updates in 2018 [2], [3]. In this operation, the Lazarus group set up fake websites and social media profiles related to cryptocurrency companies. A remarkable aspect of this operation is that Lazarus targeted *macOS* users at first. After gaining trust from the victim, Lazarus sent an email or a social media message to the victim, prompting them to install a cryptocurrency trading application.

The AppleJeus cluster updates its infection scheme continuously. When the victim installs the trojanized application, it fetches the next-stage payload after sending the victim's profiles. However, this next-stage payload is delivered selectively by the malware operator. As a result, only valuable hosts receive the following stage malware. The trojanized application fetches Manuscrypt/Fallchill variants, which are attributed to the Lazarus group. The malware also uses multi-stage infection in this phase. Generally, the installer malware implants the malware for the next phase of the infection. Eventually, the loader malware loads a backdoor component and begins backdoor operations.

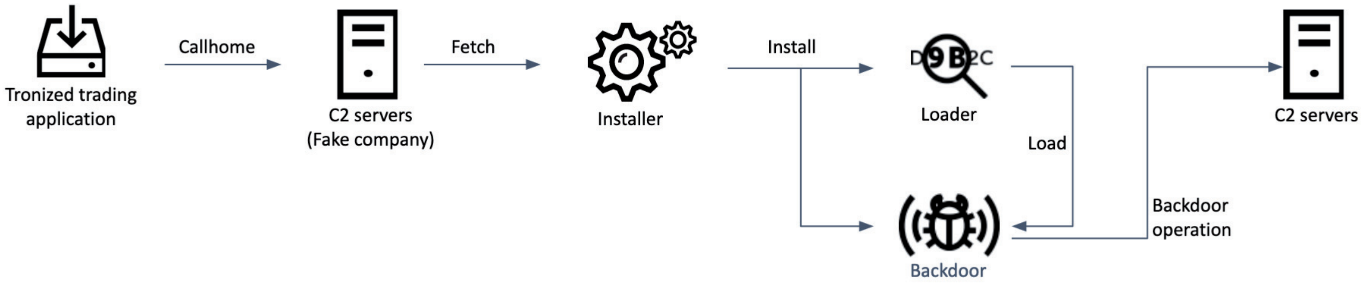


Figure 2: Example of AppleJeus infection.

From when	June 2018
Tools	Trojanized application (*Windows* and *macOS* version), installer, loader, backdoor
Targeted platform	Windows, macOS
Victimology	Cryptocurrency exchange, FinTech company

ThreatNeedle

The ThreatNeedle cluster is the most heavily used by the Lazarus group. *Kaspersky* discovered this variant in a cryptocurrency exchange in the middle of 2018. Since then, this cluster has been evolving and has been used aggressively. The Lazarus group spreads the ThreatNeedle cluster using various methods: watering-hole attacks, spear phishing with weaponized documents, and trojanized applications. ThreatNeedle is a representative malware cluster using multi-stage infection. It consists of various components, such as an installer, loader, injector, downloader and backdoor. Usually, binary infection starts from the installer or loader component after initial infection. As the name implies, the installer consists of the pieces to implant. The loader or injector is the primary element used in this cluster, and it plays a very important role. For example, when malware operators move laterally, they copy the loader malware to the remote host and execute it manually. The loader and injector always have a target file to be loaded or injected. They read the target contents from various sources:

- From the loader itself
- From a hard-coded file path
- From a registry key saved by an installer

They also use various algorithms to decrypt payloads, such as XOR, RC4 and AES. The backdoor executed by the loader and injector communicates with the C2 server and only runs in the memory. Moreover, ThreatNeedle uses various tricks to hinder analysis:

- It retrieves API addresses at runtime.
- It gets a decryption key from the command line or file.
- It loads the decryption key or configuration data from an encrypted file or registry.

During our efforts to track this cluster, we observed many victims. The Lazarus group used this cluster for various purposes: financial profit and cyber espionage. When we first noticed this cluster, it was focused on attacking cryptocurrency-related companies. However, in early 2019 it shifted targets to a mobile games developer and mobile games software company. The primary intention of the cluster is to compromise a renowned mobile application, and then collect defence and government-related intelligence using the compromised mobile application. Another big shift happened in early 2020: this cluster started to attack the defence industry aggressively [4]. We confirmed that the Lazarus group successfully compromised and exfiltrated data from one defence company. In early 2021, the *Google Threat Analysis Group* (*TAG*) announced that the

Lazarus group had attacked security researchers with highly sophisticated social engineering, and it was the ThreatNeedle cluster that was used in this campaign. As we can see, the ThreatNeedle cluster is the most widely used tool of the Lazarus group.

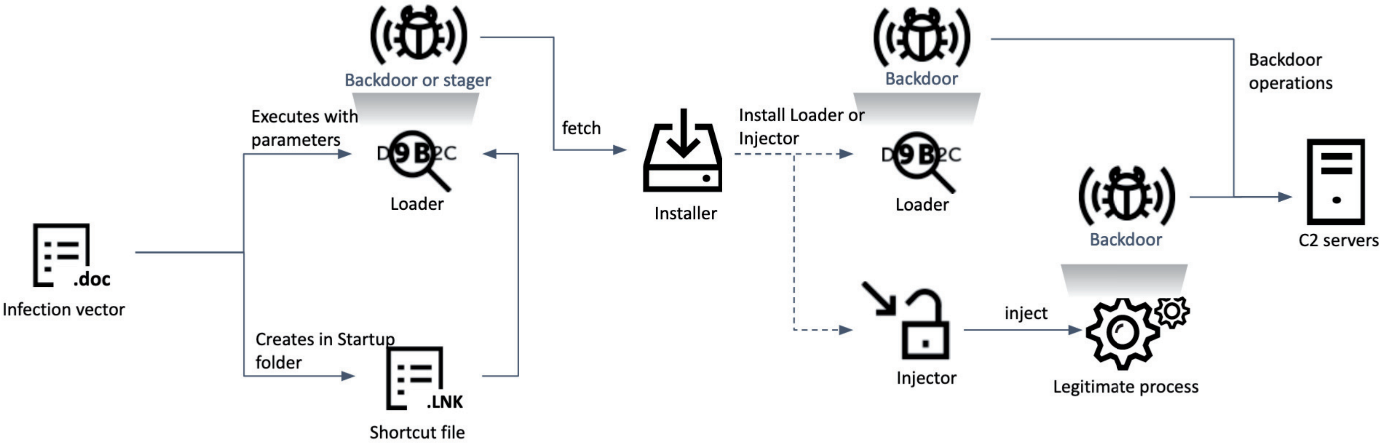


Figure 3: Example of ThreatNeedle infection.

From when	February 2018
Tools	Installer, downloader, loader, injector, backdoor
Targeted platform	*Windows*, *Android*
Victimology	Cryptocurrency businesses, mobile games company, defence industry, security researchers

DeathNote (a.k.a. DreamJob)

The DeathNote cluster was discovered in 2018 at cryptocurrency exchanges [5]. When we first found this cluster, it only targeted cryptocurrency exchanges for financial profit. But from early 2020 it shifted its target to those interested in defence industry jobs. The description of a vacancy in the defence industry was used as a decoy document. *ClearSky* also published a comprehensive 'Operation DreamJob' report on this cluster [6].

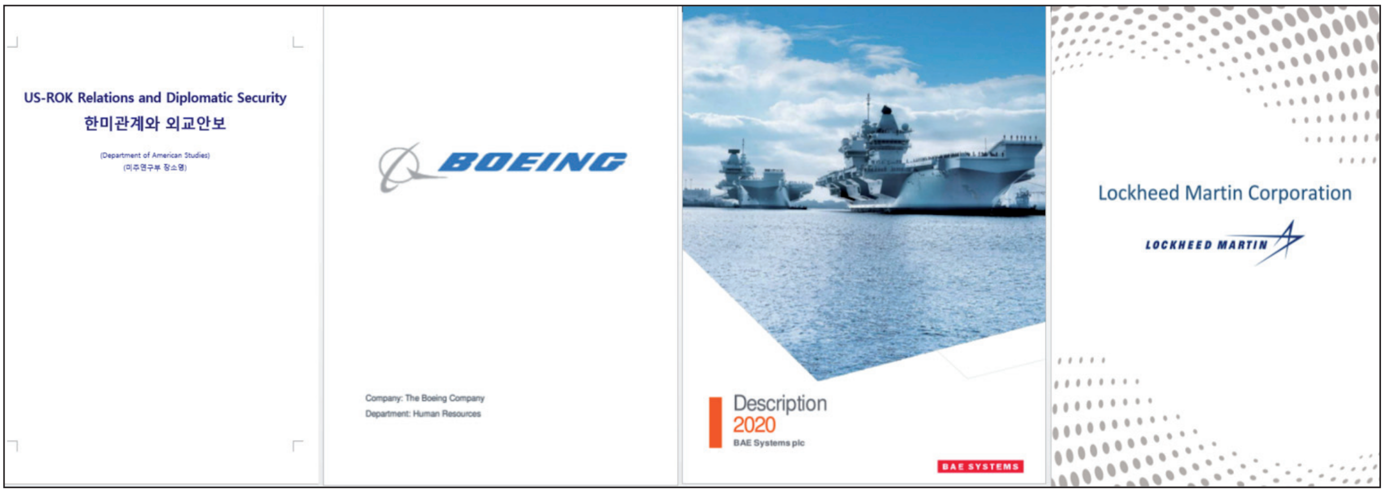

US-ROK Relations and Diplomatic Security
한미관계와 외교안보
(Department of American Studies)
(미주연구부 장소영)
BOEING
Company: The Boeing Company
Department: Human Resources
Description
2020
BAE Systems plc
BAE SYSTEMS
Lockheed Martin Corporation
LOCKHEED MARTIN

Figure 4: Decoy document related to defence industry job vacancy.

This cluster uses spear phishing with remote-template linked *Word* documents or trojanized PDF reader programs. Those two initial infection vectors create a downloader responsible for uploading profiles of the victim and fetching the next-stage payload. The malware operator only delivers the next payload if the victim is deemed valuable by the attackers. Occasionally, the threat actor uses a more complicated binary infection by adopting an injector or loader. The final payload is the Manuscrypt or COPPERHEDGE malware already attributed to the Lazarus group.

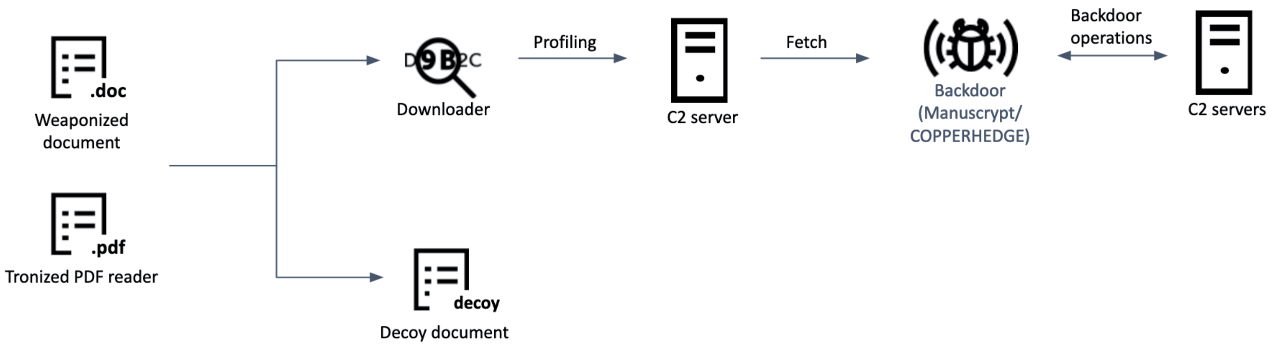


Figure 5: Example of DeathNote infection.

The DeathNote cluster has many overlaps with the ThreatNeedle cluster. Some vendors categorized the ThreatNeedle cluster as the same as the DeathNote cluster since it has also targeted the defence industry with a similar job vacancy decoy document since early 2020. However, the clusters have different payloads and techniques after the initial infection phase. Therefore, we still divide them into two distinct clusters.

From when	October 2018
Tools	Trojanized PDF reader, remote template injection malicious *Word* documents, downloader, backdoor (Manuscrypt/COPPERHEDGE)
Targeted platform	*Windows*
Victimology	Automobile, academic & defence industry

Bookcode

The Bookcode cluster is a relatively novel one. This cluster was discovered in the middle of 2020 [7], but it has been used since 2018. It utilizes the ‘bookcode’ parameter in HTTP traffic for C2 communication, hence the name Bookcode. The Lazarus group delivers this cluster using various methods: trojanized applications, watering holes, and supply-chain attacks. Initially, we discovered this malware at a software vendor in South Korea. Lazarus had delivered the Bookcode malware several times against this victim to compromise their supply chain. Moreover, *ESET* also published a report that the Lazarus group had conducted a supply-chain attack using the Bookcode cluster [8]. In early 2021, the Lazarus group also attacked a pharmaceutical company using the Bookcode cluster.

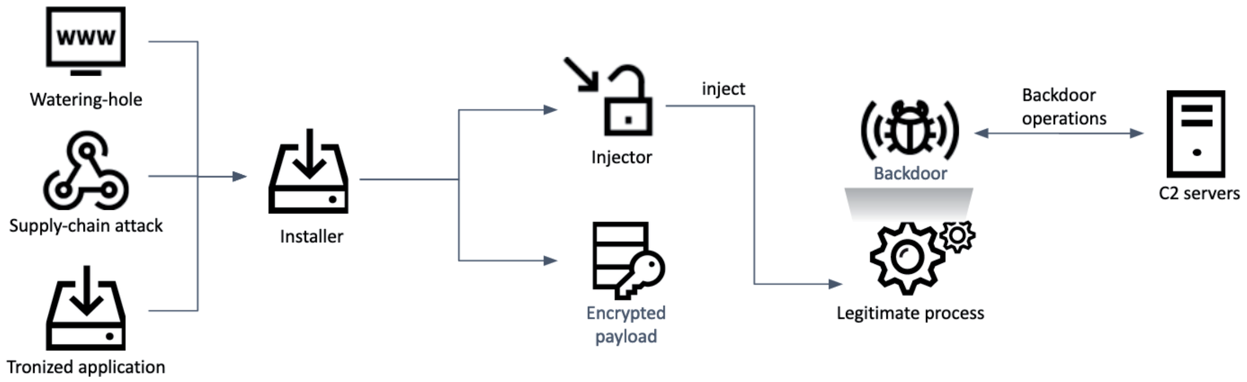


Figure 6: Example of Bookcode infection.

From when	July 2018
Tools	Installer, injector, backdoor
Targeted platform	*Windows*
Victimology	Software vendor, defence contractor, pharmaceutical company

CookieTime (a.k.a. LCPDot)

While investigating the ThreatNeedle cluster targeting the defence industry, we discovered different malware variants named CookieTime (a.k.a. LCPDot). This malware uses an encoded cookie value to deliver the request type to the C2 server and fetch command files. In this communication process, the malware takes advantage of steganography techniques. The data between the C2 server and the victim are encrypted, and a GIF header is added to evade network detection.

With this cluster, Lazarus employed macro-embedded *Word* documents or trojanized applications to infect the victim. The malicious *Word* document implants a downloader named Agamemnon (from its internal name) to fetch the next payload. After sending the profile of the victim, the CookieTime malware is delivered.

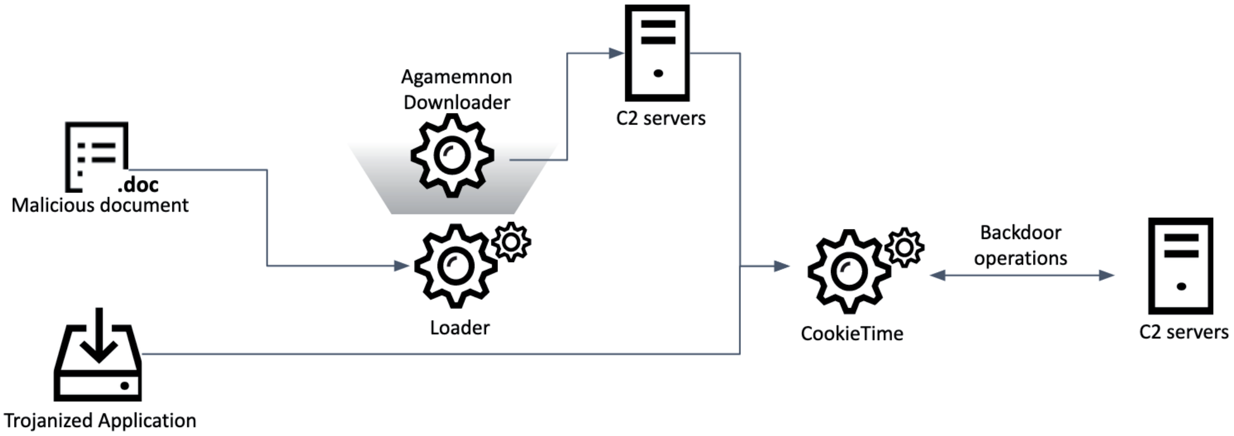


Figure 7: CookieTime infection procedure.

From when	August 2020
Tools	Trojanized application, loader, downloader, backdoor
Targeted platform	*Windows*
Victimology	Defence industry, energy industry, pharmaceutical companies

Mata (a.k.a. Dacls)

The Mata cluster has several components, such as a loader, orchestrator and plug-ins. This comprehensive framework can target *Windows*, *Linux* and *macOS* operating systems. The first artifacts we found relating to Mata were used around April 2018 [9]. After that, the actor behind this advanced malware framework used it aggressively to infiltrate corporations around the world. The primary purpose of this malware cluster was to spread ransomware or exfiltrate sensitive information from the victim. Unlike the other clusters, the Mata cluster has the same aims as a cybercrime group.

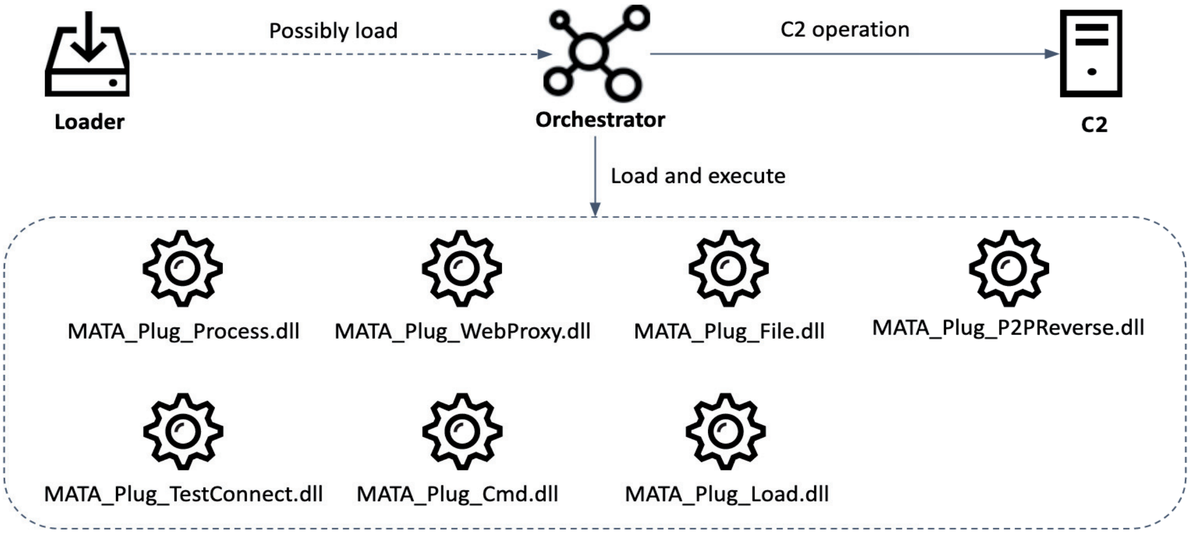


Figure 8: Mata infection procedure.

From when	May 2018
Tools	Loader, orchestrator, plug-ins
Targeted platform	*Windows*, *macOS*, *Linux*
Victimology	Various companies

CONNECTIONS BETWEEN THE CLUSTERS

Each cluster has a different infection scheme and components. However, by understanding the full attack procedure, we discovered many overlaps between the clusters. Based on these connections, we have been able to attribute these clusters to the same threat actor.

The first overlap is code similarities. Most clusters started from the same malware origin. Although the first-stage payload does not have any code overlap, the threat actor used known Lazarus malware in the second and third stages.

Another connection is the similarity of post-exploitation tools. For example, the Lazarus group used a home-made tunnelling tool to maintain auxiliary access. This tunnelling tool was discovered in both the AppleJeus cluster and the ThreatNeedle cluster.

The next overlap is the sharing of command-and-control servers. While tracking the Lazarus group, we had several opportunities to investigate its command-and-control infrastructure. The CookieTime, Bookcode and ThreatNeedle clusters made heavy use of compromised web servers in South Korea, and those clusters occasionally shared C2 servers. Although the paths of the corresponding C2 script were different, the same compromised web server was abused in a similar period. Moreover, an identical custom webshell was used to control the ThreatNeedle and DeathNote C2 servers. The Lazarus group uploads different malware cluster scripts after compromising those servers.

In addition, different malware variants were discovered during the same incident. For example, when Lazarus attacked the defence industry, it utilized a profiling malware named LPEclient with the ThreatNeedle cluster. The Bookcode cluster also used LPEclient when it attacked South Korea.

Simply put, we can summarize the connection between the clusters based on the following. Several clusters have code overlaps and similarities in metadata. Some clusters have a weak overlap with others, while in other cases one cluster has many overlaps with others. Note that ThreatNeedle, DeathNote and Bookcode have relatively strong connections. They show substantial overlaps, including sharing a C2 server, having been discovered from the same victim, and using the same tools. On the other hand, Mata has less connection with the previously known Lazarus clusters.

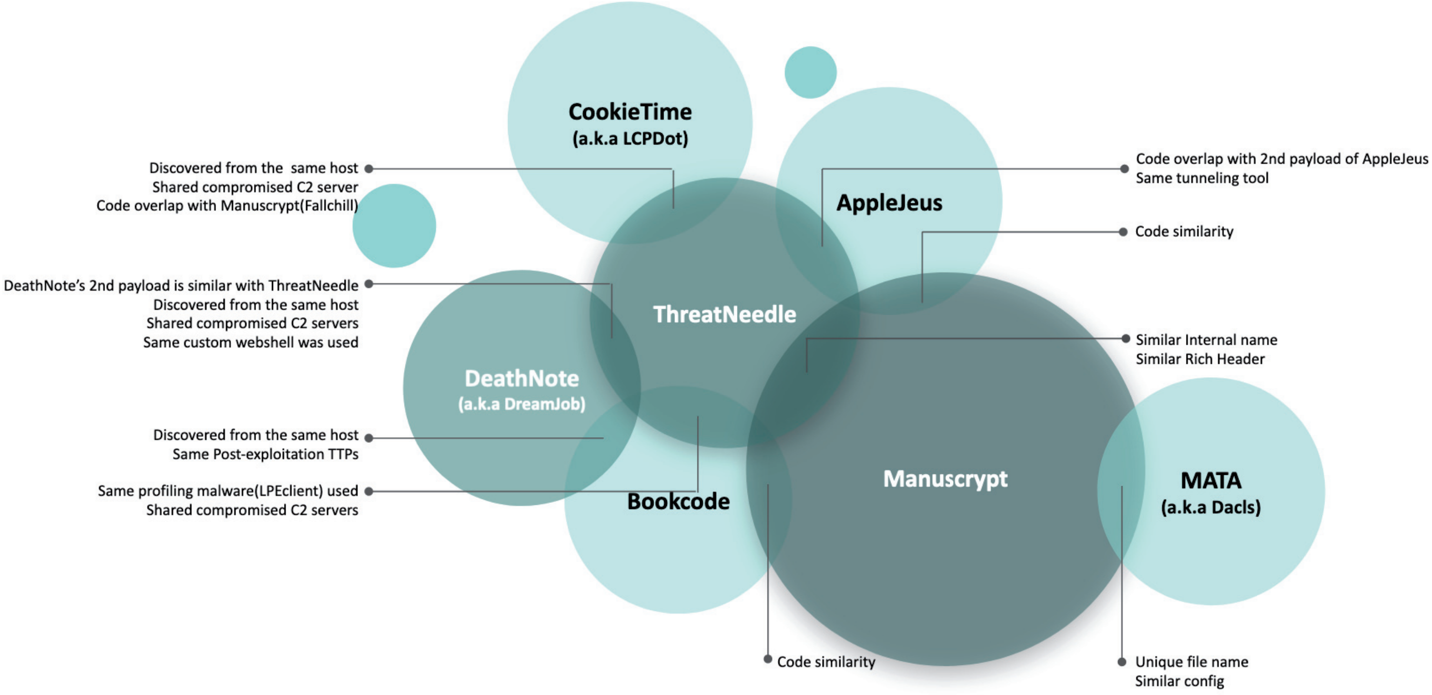

CookieTime
(a.k.a LCPDot)
AppleJeus
ThreatNeedle
DeathNote
(a.k.a DreamJob)
Bookcode
Manuscrypt
MATA
(a.k.a Dacls)
Discovered from the same host
Shared compromised C2 server
Code overlap with Manuscrypt(Fallchill)
DeathNote's 2nd payload is similar with ThreatNeedle
Discovered from the same host
Shared compromised C2 servers
Same custom webshell was used
Discovered from the same host
Same Post-exploitation TTPs
Same profiling malware(LPEclient) used
Shared compromised C2 servers
Code overlap with 2nd payload of AppleJeus
Same tunneling tool
Code similarity
Similar Internal name
Similar Rich Header
Code similarity
Unique file name
Similar config

Figure 9: Connection between clusters.

DIFFERENCES

Lazarus's arsenal has many overlaps. So why, then, do we put some tools in one cluster and call them by one name to avoid chaos in the security industry? We try to categorize those clusters because there are still many differences in their tools and techniques. Organizing each campaign based on their technical differences is essential. From the defender's perspective, different attack methodologies require different defence strategies, and the Lazarus group's clusters still have various differences. Therefore, it's necessary to cluster each cyber threat with a thorough technical background.

Used tools and techniques

When we look into each cluster's infection procedure and malware component, we can easily see that they use different infection schemes. The Lazarus group usually uses well-known initial infection vectors, some of which include spear phishing and watering holes. But several clusters prefer specific infection methods. For example, AppleJeus relied heavily on fake company websites and social media accounts to acquire a high level of trust from the victim and always generated trojanized cryptocurrency-related applications to compromise the victim. Like other threat actors, the Lazarus group relies heavily on spear phishing with a weaponized document. We've witnessed that only the ThreatNeedle and Bookcode clusters utilized the watering-hole techniques to deliver malware. The Mata cluster only used the exploitation of Internet-facing network devices.

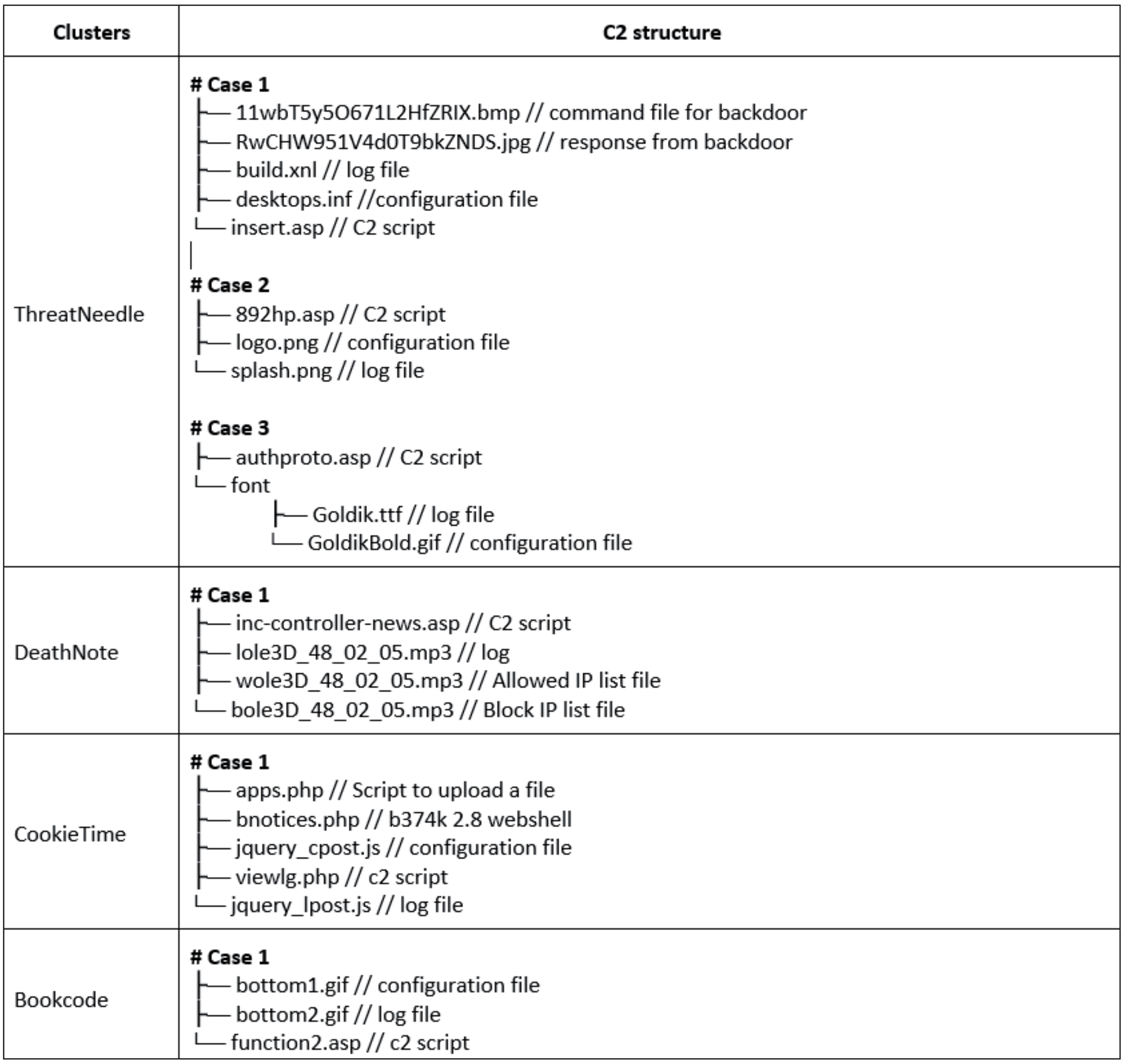
One cluster doesn't signify one malware family. Instead, each cluster consists of several components. These components are associated with the binary infection procedure, and they signify what kind of infection techniques they prefer. A relatively complicated cluster such as ThreatNeedle contains more components than others. On the other hand, the Mata cluster uses entirely different components. It works based on plug-ins, and an orchestrator controls each plug-in.

Clusters	Infection method	Malware component
AppleJeus	Trojanized application Spear phishing Contact through social media Use of Telegram channel	Installer, downloader, loader, backdoor
ThreatNeedle	Spear phishing Trojanized application (a well-known program) Watering hole Contact through social media	Installer, downloader, loader, injector, backdoor
DeathNote	Spear phishing Trojanized application (PDF reader)	Downloader, backdoor
Bookcode	Watering hole Trojanized application (a security program)	Installer, injector, backdoor
CookieTime	Spear phishing Trojanized application (a well-known program)	Downloader, loader, backdoor
Mata	Exploit vulnerable network device	Loader, orchestrator, plug-ins

Generally, the Lazarus group configures command-and-control servers in multiple stages. The first-stage C2 server is faced with the implant, and is responsible for the proxy. It saves profiles from the victim and forwards them to the second-stage server, as well as delivering the malware operator's commands from the second-stage server. The malware operator uses the second-stage server to control the first-stage servers and backdoors. Each cluster has a different C2 server structure, but the C2 servers for ThreatNeedle, Bookcode and CookieTime consist of similar components:

- C2 script: script to control implant and forward commands from the second-stage server
- Log file: saves connection logs from the implant
- Configuration file: second-stage server address

Although the structure of the C2 servers is similar, they still contain many differences. For example, while ThreatNeedle and Bookcode have similar C2 server structures, they utilize a different method when they deliver commands from the second-stage server and respond to the backdoor. The ThreatNeedle C2 server delivers that information using files, but the Bookcode C2 server uses a global variable. In addition, the DeathNote C2 server has significant differences. The DeathNote C2 server, when faced with the downloader component, has files that contain allowed and blocked IP addresses. If the conditions match, this script delivers the next-stage payload.



Clusters	C2 structure
ThreatNeedle	**# Case 1** ├── 11wbT5y5O671L2HfZRIX.bmp // command file for backdoor ├── RwCHW951V4d0T9bkZNDS.jpg // response from backdoor ├── build.xnl // log file ├── desktops.inf //configuration file └── insert.asp // C2 script │ **# Case 2** ├── 892hp.asp // C2 script ├── logo.png // configuration file └── splash.png // log file **# Case 3** ├── authproto.asp // C2 script └── font ├── Goldik.ttf // log file └── GoldikBold.gif // configuration file
DeathNote	**# Case 1** ├── inc-controller-news.asp // C2 script ├── lole3D_48_02_05.mp3 // log ├── wole3D_48_02_05.mp3 // Allowed IP list file └── bole3D_48_02_05.mp3 // Block IP list file
CookieTime	**# Case 1** ├── apps.php // Script to upload a file ├── bnotices.php // b374k 2.8 webshell ├── jquery_cpost.js // configuration file ├── viewlg.php // c2 script └── jquery_lpost.js // log file
Bookcode	**# Case 1** ├── bottom1.gif // configuration file ├── bottom2.gif // log file └── function2.asp // c2 script

Developer preference

Every software developer has a preference for a particular development environment, and it's the same for malware developers. Some malware authors prefer to develop their malware under a particular OS environment and compiler version. Many malware developers behind the Lazarus clusters still prefer Linker version 10, according to the metadata of the malware. This means they usually used Visual Studio 2010 10.0. The novel malware cluster CookieTime uses the latest version of the compiler, Visual Studio 2019 Version 16.7.

The compiler version from Rich Headers shows similar information. The fact that many malware developers behind the Lazarus clusters used various different compiler versions and heavily relied on VS2010 SP1 build 40219, suggests that different malware developers are working behind these clusters.

Clusters	Linker version from metadata	Representative compiler version from Rich Headers
AppleJeus	10, 14	VS2010 SP1 build 40219 VS2008 SP1 build 30729
ThreatNeedle	10, 14	VS2010 build 30319 VS2013 build 21005 VS2010 SP1 build 40219
DeathNote	12	VS2010 SP1 build 40219
Bookcode	10	VS2008 SP1 build 30729
CookieTime	10, 14.27	VS2010 SP1 build 40219
Mata	14.14	VS2015 UPD3.1 build 24215

When a developer develops software, they usually use a public library or source code for common functionalities. Generally, malware aims to compress the data sent to the C2 server to decrease the size. If the developer creates their code for compression, it takes a lot of time and effort. So, they usually search public codes from the Internet and adapt them for their malware. In addition, malware developers tend to prefer specific modules for covert communication. The ThreatNeedle cluster usually utilized the OpenSSL 0.9.8k library for SSL communication. However, the Mata malware developers prefer the openssl-1.1.0f library.

Clusters	zlib [10]	Curl	OpenSSL 0.9.8k	openssl-1.1.0f	SQLite Format 3
AppleJeus	X	X (macOS)			
ThreatNeedle	X		X		
DeathNote	X				X
Bookcode	X				
CookieTime					
Mata				X	

Post-exploitation tactics

Once an initial foothold has been acquired, the malware operator profiles the victim and collects additional information for further activities. The post-exploitation phase relies heavily on keyboard-hands-on activities, likely executing *Windows* commands with specific options. Thus, we can identify the habits of the humans behind the attacks from this phase. In addition, each threat actor uses a unique strategy for a post-exploitation phase.

- Generally speaking, each threat actor shows a different signature using the command line when working interactively via an installed backdoor. As a result of comparing each *Windows* commands delivered by the Lazarus group, we can confirm several differences for each cluster.
- When checking network connection with the `netstat` command, the Bookcode cluster only uses `-aon` option, rather than other clusters that use the `-ano` option.
- Only the Bookcode cluster used the `ESTA` filtering option, whereas other clusters prefer the `EST` option.
- Only the AppleJeus cluster uses the `findstr` command to filter the result rather than `find`.
- When running the `ping` command, the malware operator behind the Bookcode cluster executes it with the `-a -n 1` option. On the other hand, another operator tends to prefer to use the *`-n 1 -a`* option.

Clusters	Netstat commands	Ping commands
AppleJeus	`netstat -ano | findstr EST`	`ping -n 1 10.10.0.19`

ThreatNeedle	`netstat -ano | find "EST"`
	`netstat -ano | find ":445"`
		`ping -n 1 -a 10.10.100.100`
		`ping -n 1 -4 10.10.100.100`

DeathNote	`netstat -ano | find :445`
	`netstat -ano | find EST`
		N/A

Bookcode	`netstat -aon`
	`netstat -aon | find "ESTA"`
		`ping -a -n 1 10.10.20.67`

CookieTime	`N/A`	`ping -n 1 -a 10.1.10.192`

Mata	`netstat -ano | find "TCP"`	`ping -n 1 [host name]`
		`ping -n 1 -a 192.168.10.246`

In the post-exploitation phase, each cluster carries out a different infiltration methodology. The malware operators behind each cluster utilize different tools and techniques to achieve their ultimate goals. All of the aforementioned technical differences demonstrate why we need to separate each Lazarus cluster as much as possible; that way, those attacked can respond with the most effective defence strategy.

Clusters	Network scan	Credential access	Lateral movement	Create auxiliary access
AppleJeus				Custom port opener Custom tunnelling tool
ThreatNeedle	Wireshark	Responder [11]	SMB/Windows Admin Shares	Custom tunnelling tool
DeathNote	Custom SMB scanner	Responder [11] ChromePass [12]		
Bookcode	WakemeOnLan [13]	Responder [11] ADFind [14] BrowserPasswordRecoveryPro Manually SAM registry dump	SMB/Windows Admin Shares	
CookieTime	Custom SMB scanner	PWdump [15]	SMB/Windows Admin Shares	
Mata		Memory dumping	EternalBlue(MS17-010)	

Victimology

Every threat actor has a target. Some threat actors target geopolitical intelligence and other groups covet financial profits. The Lazarus group is one of few threat actors with extensive victimologies. Historically, the Lazarus group wanted to steal government and diplomatic intelligence. However, later on, they began to aggressively attack financial institutions for financial profits. Each cluster under the Lazarus umbrella has its own victimology.

The AppleJeus cluster has intensively attacked cryptocurrency businesses only. On the other hand, most clusters have targeted various industries. ThreatNeedle has an extensive target range and has quickly shifted its targets. ThreatNeedle first strived to obtain financial profits, but, after early 2020, it shifted its target to the defence industry. Also, surprisingly, this cluster tried to compromise security researchers in early 2021 [16]. The DeathNote cluster changed its target with the ThreatNeedle cluster. It focused on attacking cryptocurrency businesses early on, but it shifted to the defence industry from early 2020 with the ThreatNeedle cluster. ThreatNeedle and DeathNote shifted their targets in a pretty similar time frame. This means the two clusters have strong connections. The Bookcode cluster, meanwhile, heavily targeted software companies in South Korea. The operator behind the Bookcode cluster might, therefore, want to compromise the supply chain or source code. The CookieTime cluster, on the other hand, focused on the defence industry at an early stage and was then discovered attacking a pharmaceutical company when countries were developing the Covid-19 vaccine competitively [17]. Unlike other clusters, Mata has different victimologies because its nature is similar to that of cybercrime groups. Mata compromised small and medium-sized companies all over the globe and attempted to steal sensitive data related to their customers or to spread homemade ransomware.

	AppleJeus	ThreatNeedle	DeathNote	Bookcode	CookieTime	Mata
Cryptocurrency businesses	X	X	X			
Defence		X	X		X	
S/W company		X		X		
Security researchers		X				
Pharmaceutical companies				X	X	
Cybercrime-like target						X

One more difference in victimology is the range of target countries. The Lazarus group has attacked South Korea primarily because of geopolitical reasons. But some clusters attacked numerous countries around the world, excluding South Korea. While the Bookcode cluster was only discovered in South Korea, the AppleJeus, and Mata clusters attacked countries around the world except for South Korea. ThreatNeedle, DeathNote and CookieTime are not limited to any specific country.

- South Korea only: Bookcode
- Global excluding South Korea: AppleJeus, Mata
- Global including South Korea: ThreatNeedle, DeathNote, CookieTime

CONCLUSION

Any group or organization that we come across in our daily life has the ability to change its form and membership. Highly skilled threat actors, even state-sponsored groups, follow these simple principles. They may change internal members, structure, or even leadership and, in doing so, they create changes in their capabilities and methodology. The notorious Lazarus group has been active for a decade, and it's apparent that the group has undergone many changes internally during this period. In addition, active threat actors such as the Lazarus tend to evolve their tools and adopt novel technologies aggressively. This means that the longer known threat groups are active, the harder it becomes to attribute new clusters of activity to them, given that they have often retooled and reorganized old tools.

Each security vendor, even each threat intelligence analyst, has a different standard for attribution. Despite the origin of the cyber attack being the same, each vendor can generate a different conclusion. As the scale and sophistication of cyber attacks by one group become greater and the changes more frequent, the gaps between vendors are increased. Lazarus group is an excellent example of this. When the security industry first started to shed light on this group, its was relatively small and didn't have many clusters. But, now, it is a major threat actor with sufficient capabilities for attacking various industries and with a highly sophisticated set of skills. Many security vendors have endeavoured to classify the group's various campaigns but, in doing so, the gaps between security vendors' understandings have become clear. Lazarus is not likely to disappear from the threat landscape anytime soon, which means attribution will continue to be a complicated issue. By explaining our process for classifying various Lazarus clusters, we aim to close the gaps in individual vendors' knowledge. We need to be careful when drawing conclusions and share each other's perspectives with an open mind. Also, we need to look back at what we have discovered in the past to determine what we might have missed and reduce the difference.

REFERENCES

[1] Operation Blockbuster: https://operationblockbuster.com/.

[2] Kaspersky. Operation AppleJeus: https://securelist.com/operation-applejeus/87553/.

[3] Kaspersky. Operation AppleJeus Sequel: https://securelist.com/operation-applejeus-sequel/95596/.

[4] Kaspersky. Lazarus targets defense industry with ThreatNeedle: https://securelist.com/lazarus-threatneedle/100803/.

[5] JPCERT. Operation Dreamjob by Lazarus: https://blogs.jpcert.or.jp/en/2021/01/Lazarus_malware2.html.

[6] ClearSky. Operation DreamJob: https://www.clearskysec.com/operation-dream-job/.

[7] KISA. Analysis of cases of control of internal network through TTPs#1. https://www.krcert.or.kr/data/reportView.do?bulletin_writing_sequence=35330.

[8] ESET. Lazarus supply-chain attack in South Korea: https://www.welivesecurity.com/2020/11/16/lazarus-supply-chain-attack-south-korea/.

[9] Kaspersky. MATA: Multi-platform targeted malware framework: https://securelist.com/mata-multi-platform-targeted-malware-framework/97746/.

[10] Zlib. https://zlib.net/.

[11] https://github.com/lgandx/Responder/.

[12] https://www.nirsoft.net/utils/chromepass.html.

[13] https://www.nirsoft.net/utils/wake_on_lan.html.

[14] http://www.joeware.net/freetools/tools/adfind/.

[15] https://www.openwall.com/passwords/windows-pwdump.

[16] Google TAG. New campaign targeting security researchers: https://blog.google/threat-analysis-group/new-campaign-targeting-security-researchers/.

[17] Kaspersky. Lazarus covets COVID-19-related intelligence: https://securelist.com/lazarus-covets-covid-19-related-intelligence/99906/.

[18] McAfee. Malbus: https://www.mcafee.com/blogs/other-blogs/mcafee-labs/malbus-popular-south-korean-bus-app-series-in-google-play-found-dropping-malware-after-5-years-of-development/.

SAMPLES HASH

A part of each cluster for internal research.

AppleJeus

0bdb652bbe15942e866083f29fb6dd62 Trojanized application

bbbcf6da5a4c352e8846bf91c3358d5c Downloader

d7089e6bc8bd137a7241a7ad297f975d Backdoor

ThreatNeedle

69da2c56a56fecb981e326cb6ea42704 Loader

c34d5d2cc857b6ee9038d8bb107800f1 Loader

e441f021b1c8a3d481be0a5312378d6f Installer

4c1d8c4142f2a260f69ec8d597ba51fa Backdoor

140a5572e0171cfe393321017b9cdee9 Backdoor

DeathNote (a.k.a. DreamJob)

c04e50275ab9c4b22f39bcd61db0da76 Trojanized PDF reader

d1c652b4192857cb08907f0ba1790976 Downloader

7228705813d5bc6c6a62fc53ac019344 Downloader

Bookcode

3d0355ff78dcc979b3f83a679b6ba794 Backdoor

74b16e70e721cdb6cd04fc8220c93dd2 Backdoor

ddf6bd6ad5e40b236492d06e40d197ca Backdoor

CookieTime (a.k.a. LCPDot)

06adca7a28b6d1d983912f7f544ee413 Trojanized application

d59a0a04abcb38fdb391a09972aa3ff4 Backdoor

b8df94ce84201b17684e0d368ed38024 Backdoor

05ae0af44b62f4df432b281809e90f67 Backdoor

Mata (a.k.a.Dacls)

859e7e9a11b37d355955f85b9a305fec Linux Mata

7b068dfbea310962361abf4723332b3a Loader

da50a7a05abffb806f4a60c461521f41 Windows Orchestrator

ec05817e19039c2f6cc2c021e2ea0016 Windows Orchestrator