비전문가도 활용가능한 서비스형 악성코드
국내 K기관 22GB분량의 내부문서 다크웹에 유포

Lummar Stealer 분석

요약

1. Lummar Stealer는 C언어 기반의 정보탈취 악성코드로, 서비스형 악성코드 모델을 통해 유포됨.
2. 감염된 PC 내 암호화폐 지갑, 브라우저 내 저장된 계정정보, 결제정보 탈취가 주 목적
3. 피해사례:

피해기업	피해 내용
국내 K 기관	·개인식별 및 사칭 가능성이 높은 문서 등 22GB 분량 데이터 유출. ·다크웹에서 15,000달러에 판매.

4. 주요공격 기술

공격 단계	기술 ID	설명
방어 회피 (Defense Evasion)	T1027	파일 암호화 및 난독화
	T1027.013	암호화/인코딩된 파일 사용
정보 수집 (Collection)	T1119	자동화된 정보 수집
	T1217	브라우저 정보 탐색
자격 증명 접근 (Credential Access)	T1555.003	브라우저에 저장된 비밀번호 탈취
명령 및 제어 (Command and Control)	T1017.001	웹 프로토콜을 이용한 통신
유출 (Exfiltration)	T1573	암호화된 채널 사용
	T1041	C2 채널을 통한 데이터 유출

대응 방안

1. MS에서 제공하는 보안 업데이트를 자동으로 설정한다.
2. 사용 중인 소프트웨어의 최신 업데이트 상태를 유지한다.
3. 백신 최신 업데이트 상태를 유지한다.
4. 주요 문서는 주기적으로 백업하고 물리적으로 분리하여 관리한다.
5. 신뢰할 수 없는 메일의 첨부파일은 실행을 금지한다.
6. 비 업무 사이트 및 신뢰할 수 없는 웹사이트의 연결을 차단한다.

목차

1. 개요

1.1 배경

[그림 1] Lumma Stealer 로고

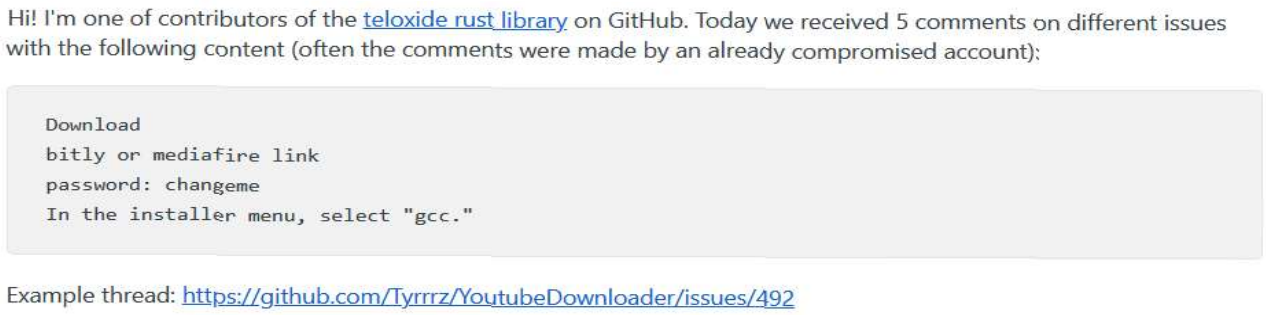
C언어 기반의 정보 탈취 악성코드인 Lumma Stealer(LummaC2)는
최소 2022년 8월부터 러시아어권 포럼에서
서비스형 악성코드(MaaS, Malware-as-a-Service) 모델을 통해 유포된 것으로 확인되었다.

MaaS란 악성코드 수요자에게 즉시 배포 가능한 멀웨어를 제공하는 비즈니스 모델로,
이러한 방식은 전문 기술이 없는 일반 사용자도
쉽게 사이버 공격을 감행할 수 있게 만든다는 점에서 위협적이다.

빌드 크기는 약 150~396KB이며 스틸러의 가격은 세 가지 범주로 나뉜다.

- **Experienced – $250/month:**
  대량의 로그를 다운로드하고 특정 매개변수로 로그를 필터링하는 기능을 포함
- **Professional – $500/month:**
  스틸러에 대한 무제한 규칙생성, 다른 사람과 통계 공유, 비상주 로더 등의 기능을 포함
- **Corporate – $1000/month:**
  사용자가 무작위 빌드를 받아서 생성된 빌드는 다운받는 사람마다 서로 다르며
  Heaven's Gate 기법이 구현되어 동적 분석을 우회하는 기능 포함

Lumma Stealer의 주된 공격 목표는
감염된 PC로부터 암호화폐 지갑 주소, 웹 브라우저에 저장된 비밀번호와 쿠키,
신용카드 정보, 은행 정보 및 기타 중요한 개인정보를 탈취하는 것이다.


Hi! I'm one of contributors of the teloxide rust library on GitHub. Today we received 5 comments on different issues with the following content (often the comments were made by an already compromised account):

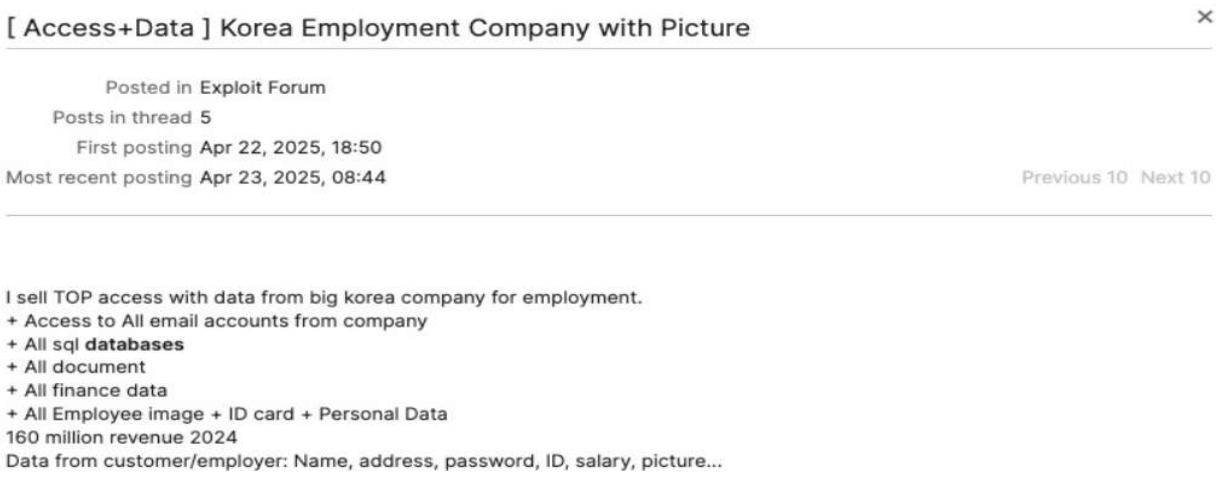
```
Download
bitly or mediafire link
password: changeme
In the installer menu, select "gcc."
```

Example thread: https://github.com/Tyrrrz/YoutubeDownloader/issues/492

[그림 2] GitHub 프로젝트 댓글의 Lumma Stealer 피싱 링크

2024년에는 새로운 유포 방식이 확인되었다.
공격자들은 GitHub 프로젝트 댓글에 가짜 수정 파일(fix)로 위장한 ZIP 아카이브를 게시하여,
사용자가 직접 Lumma Stealer를 실행하도록 유도했다.

또한 가짜 CAPTCHA 페이지와 'ClickFix' 기법을 이용해
악성코드를 다운로드하게 만드는 수법도 사용되었다.


[Access+Data] Korea Employment Company with Picture

Posted in Exploit Forum
Posts in thread 5
First posting Apr 22, 2025, 18:50
Most recent posting Apr 23, 2025, 08:44
Previous 10 Next 10

I sell TOP access with data from big korea company for employment.
+ Access to All email accounts from company
+ All sql **databases**
+ All document
+ All finance data
+ All Employee image + ID card + Personal Data
160 million revenue 2024
Data from customer/employer: Name, address, password, ID, salary, picture...

[그림 3] 다크웹 해킹 포럼의 K사 데이터 유출 게시글

최근 2025년 4월, 국내 K기관이 Lumma Stealer에 감염되어 내부 데이터가 유출되는 사고가 발생했다.

유출된 데이터는 약 22GB 분량으로, 다크웹에서 15,000달러에 판매 중인 것으로 확인되었다.
해당 게시글에 따르면, 유출된 자료에는 일부 암호화되지 않은 개인 정보와 함께
개인 식별 및 신분 도용에 악용될 소지가 큰 문서들이 포함된 것으로 알려졌다.

2. 분석

2.1 파일 정보

- PE 파일(SHA-256)

Name	LummaC2.4.exe
Type	Windows 실행 파일
Behavior	Infostealer
SHA-256	7a5fe28d6e4be7999b697c92c963098f660bbcd9bc89841a1cd9836f22a4266c
Description	Lumma Stealer(LummaC2 Stealer)
File Size	334.00 KB

- C2 Config 페이로드

Name	output.241471247.txt
Type	txt 파일
SHA-256	533a6854efe4f57824ad9e8154d448d22eab66af085e377e190b648f2363ecef
File Size	7.27 KB

- C2 서버

- inspirzedthoughts.tech
- pasteflawwed.world
- hoyoverse.blog
- dsfljsdfjewf.info
- stomlegue.com
- blast-hubs.com
- blastikcn.com
- decreaserid.world
- lestagames.world
- hxxps://steamcommunity.com/profiles/76561199822375128

2.2 MITRE ATT&CK

Command and Control

- (T1017.001) Application Layer Protocol: Web Protocols
- (T1573) Encrypted Channel

Collection

- (T1119) Automated Collection

Discovery

- (T1217) Browser Information Discovery
- (T1082) System Information Discovery

Credential Access

- (T1555.003) Credentials from Password Stores

Exfiltration

- (T1041) Exfiltration Over C2 Channel

Defense Evasion

- (T1027) Obfuscated Files or Information
- (T1027.013) Obfuscated Files or Information: Encrypted/Encoded File

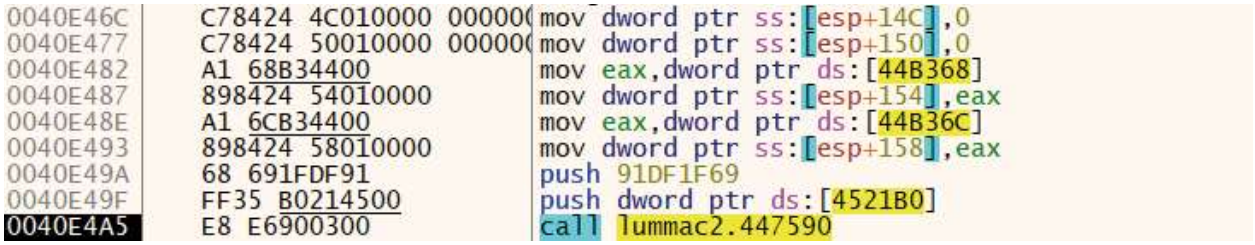
3. 분석

3.1 탐지회피

Lumma Stealer는 탐지 회피 및 분석 지연을 위해
문자열 암호화/난독화, API Resolving 기법 등을 사용한다.

3.1.1 문자열 암호화

해당 악성코드(Lumma Stealer)는 내부 C2 주소를
chacha20 알고리즘으로 암호화하여 내부에 가지고 있다.



[그림 4] 암호화 되어 있는 문자열 복호화

대칭키 암호화인 chacha20 알고리즘으로 암호화된 API 문자열을 복호화 한다.

오프셋	크기	설명
0019FAA0	16	chacha20에 사용되는 16바이트 문자열
+ 0x10(16)	32	chacha20 대칭키
+ 0x8	8	chacha20 카운터 값
+ 0x8	8	chacha20 Nonce값

[표 1] chacha20 키 스트림

3.1.2 API Resolving

Lumma stealer는 분석 지연 및 패턴 기반의 탐지를 회피하기 위해 API Resolving을 사용한다.
API Resolving이란, 프로그램이 함수를 이름으로 직접 호출하는 대신
실행 시점에 필요한 함수의 실제 주소를 동적으로 찾아 호출하는 기법이다.

이 기법은 본래 호환성 확보나 의존성 최소화 등을 위해 정상 프로그램에서도 사용되지만,
악성코드의 경우 주로 정적 분석을 무력화하고 탐지 시스템을 우회하기 위한 수단으로 이를 악용한다.

```
0040E4F5    FFB424 5C010000    push dword ptr ss:[esp+15C]
0040E4FC    FF35 B0214500      push dword ptr ds:[4521B0]
0040E502    E8 89900300        call lummac2.447590
0040E507    83C4 08            add esp,8
0040E50A    A3 78214500        mov dword ptr ds:[<&WinHttpCloseHandle>],eax
0040E50F    68 9198F6B7        push B7F69891
0040E514    FF35 B0214500      push dword ptr ds:[4521B0]
0040E51A    E8 71900300        call lummac2.447590
0040E51F    83C4 08            add esp,8
0040E522    A3 7C214500        mov dword ptr ds:[<&WinHttpConnect>],eax
0040E527    68 EA658B9B        push 9B8B65EA
0040E52C    FF35 B0214500      push dword ptr ds:[4521B0]
0040E532    E8 59900300        call lummac2.447590
0040E537    83C4 08            add esp,8
0040E53A    A3 80214500        mov dword ptr ds:[<&WinHttpCrackUrl>],eax
0040E53F    68 811E579A        push 9A571E81
0040E544    FF35 B0214500      push dword ptr ds:[4521B0]
0040E54A    E8 41900300        call lummac2.447590
0040E54F    83C4 08            add esp,8
0040E552    A3 84214500        mov dword ptr ds:[<&WinHttpOpen>],eax
0040E557    A1 B4D34400        mov eax,dword ptr ds:[44D3B4]
0040E55C    B9 7D676624        mov ecx,2466677D
0040E561    330D BCD34400      xor ecx,dword ptr ds:[44D3BC]

0040DFDC    68 569D51C5        push C5519D56
0040DFE1    FF35 B0214500      push dword ptr ds:[4521B0]
0040DFE7    E8 A4950300        call lummac2.447590
0040DFEC    83C4 08            add esp,8
0040DFEF    A3 88214500        mov dword ptr ds:[<&WinHttpOpenRequest>],eax
0040DFF4    68 5AC6176A        push 6A17C65A
0040DFF9    FF35 B0214500      push dword ptr ds:[4521B0]
0040DFFF    E8 8C950300        call lummac2.447590
0040E004    83C4 08            add esp,8
0040E007    A3 8C214500        mov dword ptr ds:[<&WinHttpQueryDataAvailable>]
0040E00C    68 9DA6C4C1        push C1C4A69D
0040E011    FF35 B0214500      push dword ptr ds:[4521B0]
0040E017    E8 74950300        call lummac2.447590
0040E01C    83C4 08            add esp,8
0040E01F    A3 90214500        mov dword ptr ds:[<&WinHttpReadData>],eax
0040E024    68 8DF2FC04        push 4FCF28D
0040E029    FF35 B0214500      push dword ptr ds:[4521B0]
0040E02F    E8 5C950300        call lummac2.447590
0040E034    83C4 08            add esp,8
0040E037    A3 94214500        mov dword ptr ds:[<&WinHttpReceiveResponse>],ea
0040E03C    68 08F41FAB        push AB1FF408
0040E041    FF35 B0214500      push dword ptr ds:[4521B0]
0040E047    E8 44950300        call lummac2.447590
0040E04C    83C4 08            add esp,8
0040E04F    A3 98214500        mov dword ptr ds:[<&WinHttpSendRequest>],eax
0040E054    68 1F149C8C        push 8C9C141F
0040E059    FF35 B0214500      push dword ptr ds:[4521B0]
0040E05F    E8 2C950300        call lummac2.447590
0040E064    83C4 08            add esp,8
0040E067    A3 9C214500        mov dword ptr ds:[<&WinHttpSetTimeouts>],eax
0040E06C    68 3C6BAD5E        push 5EAD6B3C
0040E071    FF35 B0214500      push dword ptr ds:[4521B0]
0040E077    E8 14950300        call lummac2.447590
0040E07C    83C4 08            add esp,8
0040E07F    A3 A0214500        mov dword ptr ds:[<&WinHttpWriteData>],eax
```

[그림 5] winhttp.dll API Resolving

Lumma Stealer는 먼저 447590 함수를 호출하여
winhttp.dll에서 사용할 API의 주소를 계산하고 이를 메모리에 할당한다.
그 외 다른 API들은 XOR 연산을 통해 주소를 알아내며,
이렇게 가져온 주요 DLL과 API 목록은 [표 2]와 같다.

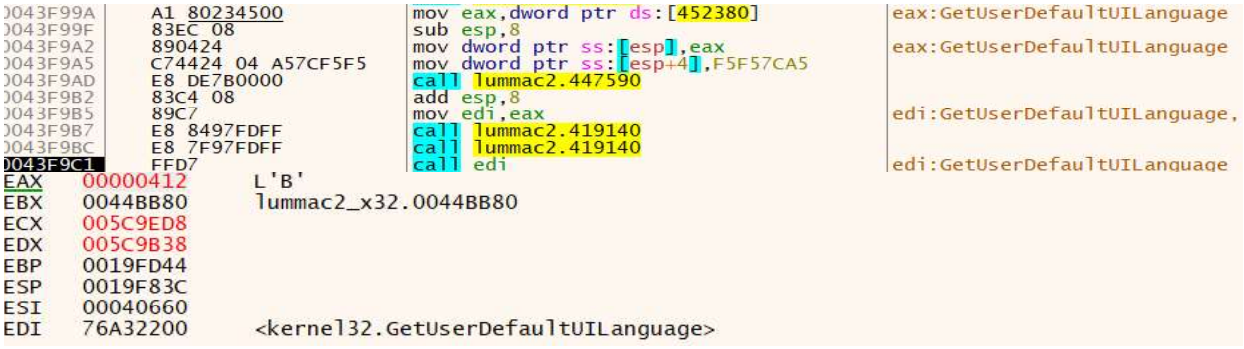
KERNEL32.DLL
LoadLibraryExW, FreeLibrary
ntdll.dll
RtlAllocateHeap, RtlReAllocateHeap, RtlHeapFree

[표 2] API Resolving

3.1.3 패킹 여부 확인

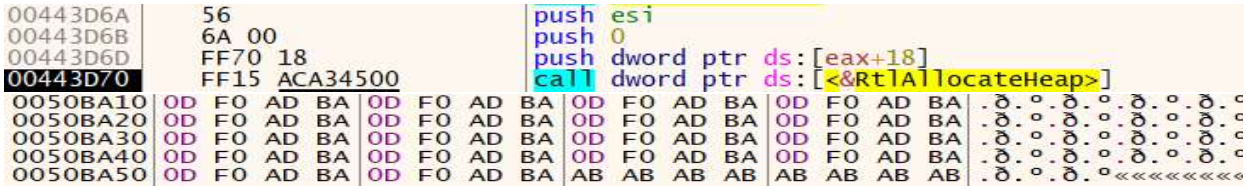
Lumma Stealer 4.0 버전에는 새로운 기능이 추가되었다.
패킹(packing)되지 않은 원본 악성코드를 실행하면,
해당 파일이 악성코드임을 경고하며 실행 여부를 묻는 확인 창이 나타난다.

개발자 측은 이 기능이 암호화되지 않은 원본 샘플의 유출을 방지하기 위한 조치라고 밝혔다.



[그림 6] GetUserDefaultUILanguage

GetUserDefaultUILanguage API를 호출하여 현재 실행 중인 PC의 언어를 확인한다.
EAX 레지스터에 반환값인 0x412가 들어왔으므로 한국어를 사용 중인 환경임을 알 수 있다.



[그림 7] RtlAllocateHeap

RtlAllocateHeap API로 힙에 메모리 블록을 할당한다.

```
00448CB0    66:8911             mov word ptr ds:[ecx],dx
00448CB3    83C1 02             add ecx,2
00448CB6    0FB713              movzx edx,word ptr ds:[ebx]
00448CB9    83C3 02             add ebx,2
00448CBC    66:85D2             test dx,dx
00448CBF  ^ 75 EF               jne lummac2.448CB0
00448CC1    66:C701 0000        mov word ptr ds:[ecx],0
00448CC6    0FB716              movzx edx,word ptr ds:[esi]
00448CC9    66:85D2             test dx,dx
00448CCC  v 74 23               je lummac2.448CF1
00448CCE    83C6 02             add esi,2
00448CD1    90                  nop
00448CD2    90                  nop
00448CD3    90                  nop
00448CD4    90                  nop
00448CD5    90                  nop
00448CD6    90                  nop
00448CD7    90                  nop
00448CD8    90                  nop
00448CD9    90                  nop
00448CDA    90                  nop
00448CDB    90                  nop
00448CDC    90                  nop
00448CDD    90                  nop
00448CDE    90                  nop
00448CDF    90                  nop
00448CE0    66:8911             mov word ptr ds:[ecx],dx
00448CE3    83C1 02             add ecx,2
00448CE6    0FB716              movzx edx,word ptr ds:[esi]
00448CE9    83C6 02             add esi,2
00448CEC    66:85D2             test dx,dx
00448CEF  ^ 75 EF               jne lummac2.448CE0
0050BA10 5C 00 3F 00 3F 00 5C 00 43 00 3A 00 5C 00 55 00 \.?.?.\.C.:.\.U.
0050BA20 73 00 65 00 72 00 73 00 5C 00 50 00 43 00 5C 00 s.e.r.s.\.P.C.\.
0050BA30 44 00 65 00 73 00 6B 00 74 00 6F 00 70 00 5C 00 D.e.s.k.t.o.p.\.
0050BA40 4C 00 75 00 6D 00 6D 00 61 00 43 00 32 00 2E 00 L.u.m.m.a.C.2...
0050BA50 65 00 78 00 65 00 AD BA AB AB AB AB AB AB AB AB e.x.e..º«««««««««
```

[그림 8] Lumma stealer 실행 경로

앞서 할당한 메모리 블록에 현재 Lumma stealer가 실행되고 있는 경로를 가져온다.

```
004458B0    8B4424 04           mov eax,dword ptr ss:[esp+4]
004458B4    897424 FC           mov dword ptr ss:[esp-4],esi
004458B8    897C24 F8           mov dword ptr ss:[esp-8],edi
004458BC    8B4C24 08           mov ecx,dword ptr ss:[esp+8]
004458C0    8D7424 0C           lea esi,dword ptr ss:[esp+C]
004458C4    8D7C24 04           lea edi,dword ptr ss:[esp+4]
004458C8    F3:A4               rep movsb
004458CA    8B7424 FC           mov esi,dword ptr ss:[esp-4]
004458CE    8B7C24 F8           mov edi,dword ptr ss:[esp-8]
004458D2    8D5424 04           lea edx,dword ptr ss:[esp+4]
004458D6    FF15 B8A34500       call dword ptr ds:[45A3B8]
76EA7000    EA 0970EA76 3300    jmp far 33:76EA7009
76EA7007    0000                add byte ptr ds:[eax],al
76EA7009    41                  inc ecx
76EA700A  - FFA7 F8000000       jmp dword ptr ds:[edi+F8]
```

[그림 9] Heaven's gate

4458D6번지에서 동적 분석을 방해하기 위해 Heaven's gate 기법을 사용한다.

```
win32u!NtUserGetForegroundWindow:
76e813c0 b83c100100 mov     eax, 1103Ch
76e813c5 ba2063e876 mov     edx, 76E86320h
76e813ca ffd2       call    edx
76e813cc c3         ret
```

[그림 10] NtUserGetForegroundWindow

Heaven's gate 기법으로 NtUserGetForegroundWindow API를 호출하여
GUI에서 현재 실행중인 최상단 창의 정보를 가져온다.

```
00443D98    56                  push esi
00443D99    6A 00               push 0
00443D9B    FF70 18             push dword ptr ds:[eax+18]
00443D9E    FF15 B4A34500       call dword ptr ds:[<&RtlFreeHeap>]
0050BA10  90 B6 50 00  F0 AB 50 00  EE FE EE FE  EE FE EE FE  .¶P.ð«P.îþîþîþîþ
0050BA20  EE FE EE FE  EE FE EE FE  EE FE EE FE  EE FE EE FE  îþîþîþîþîþîþîþîþ
0050BA30  EE FE EE FE  EE FE EE FE  EE FE EE FE  EE FE EE FE  îþîþîþîþîþîþîþîþ
0050BA40  EE FE EE FE  EE FE EE FE  EE FE EE FE  EE FE EE FE  îþîþîþîþîþîþîþîþ
0050BA50  EE FE EE FE  EE FE EE FE  EE FE EE FE  EE FE EE FE  îþîþîþîþîþîþîþîþ
```

[그림 11] RtlFreeHeap

RtlFreeHeap으로 Lumma stealer의 실행 경로가 담긴 메모리 블록을 해제한다.

```
00443D6A    56              push esi
00443D6B    6A 00           push 0
00443D6D    FF70 18         push dword ptr ds:[eax+18]
00443D70    FF15 ACA34500   call dword ptr ds:[<&RtlAllocateHeap>]
0051DEE0  0D F0 AD BA 0D F0 AD BA 0D F0 AD BA 0D F0 AD BA  .ð.º.ð.º.ð.º.ð.º
0051DEF0  0D F0 AD BA 0D F0 AD BA 0D F0 AD BA 0D F0 AD BA  .ð.º.ð.º.ð.º.ð.º
0051DF00  0D F0 AD BA 0D F0 AD BA 0D F0 AD BA 0D F0 AD BA  .ð.º.ð.º.ð.º.ð.º
0051DF10  0D F0 AD BA 0D F0 AD BA 0D F0 AD BA 0D F0 AD BA  .ð.º.ð.º.ð.º.ð.º
0051DF20  0D F0 AD BA 0D F0 AD BA 0D F0 AD BA 0D F0 AD BA  .ð.º.ð.º.ð.º.ð.º
0051DF30  0D F0 AD BA 0D F0 AD BA 0D F0 AD BA 0D F0 AD BA  .ð.º.ð.º.ð.º.ð.º
0051DF40  0D F0 AD BA 0D F0 AD BA 0D F0 AD BA 0D F0 AD BA  .ð.º.ð.º.ð.º.ð.º
0051DF50  0D F0 AD BA 0D F0 AD BA 0D F0 AD BA 0D F0 AD BA  .ð.º.ð.º.ð.º.ð.º
0051DF60  0D F0 AD BA 0D F0 AD BA 0D F0 AD BA 0D F0 AD BA  .ð.º.ð.º.ð.º.ð.º
```

[그림 12] 재할당

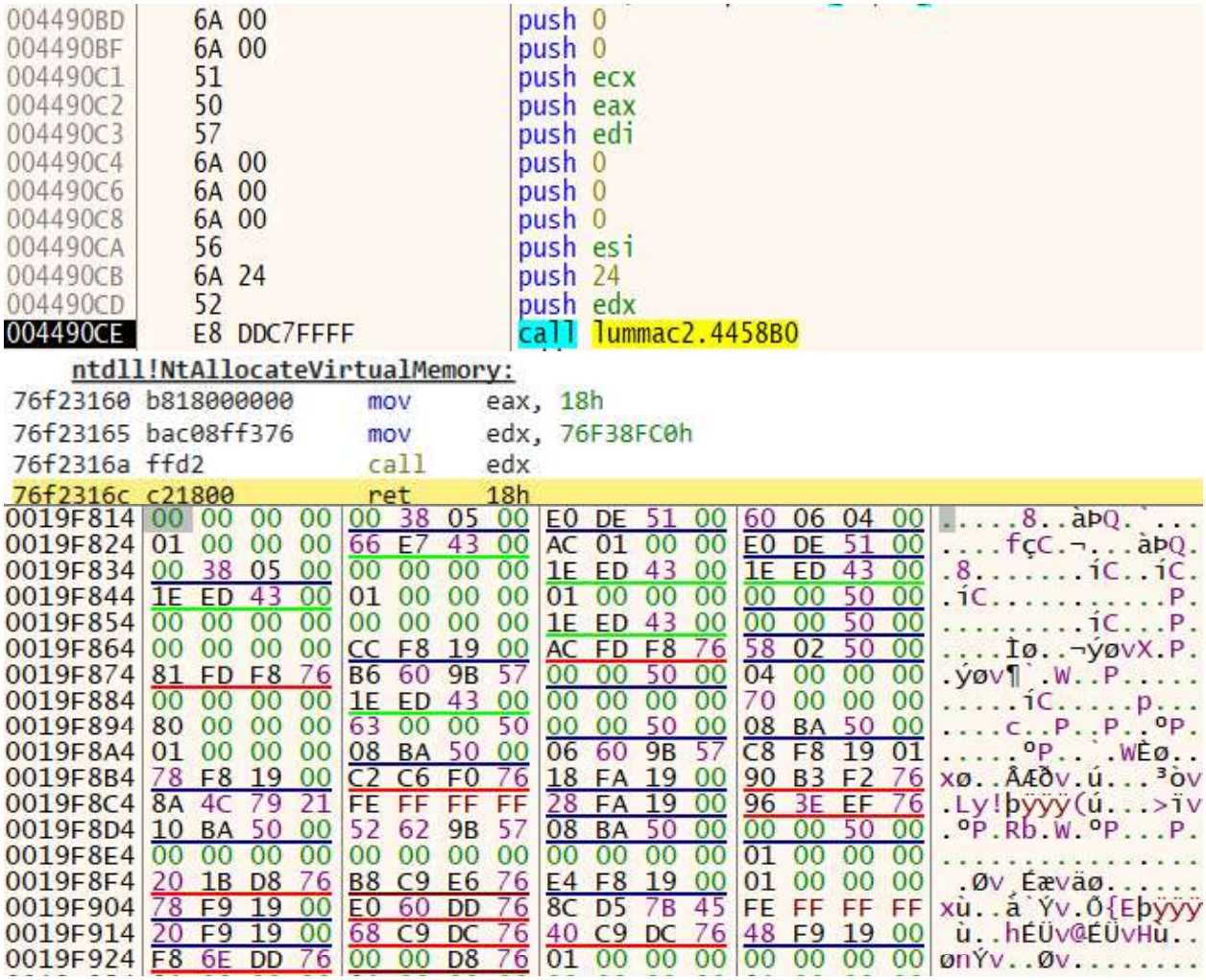
다시 RtlAllocateHeap API로 힙에 메모리 블록을 할당한다.

```
0043E92C    898D FCFDFFFF   mov dword ptr ss:[ebp-204],ecx
0043E932    8B8D F8FDFFFF   mov ecx,dword ptr ss:[ebp-208]
0043E938    8B95 FCFDFFFF   mov edx,dword ptr ss:[ebp-204]
0043E93E    8B0C91          mov ecx,dword ptr ds:[ecx+edx*4]
0043E941    8B15 F8124500   mov edx,dword ptr ds:[4512F8]
0043E947    81F2 920B8FC8   xor edx,C88F0B92
0043E94D    29D1            sub ecx,edx
0043E94F    FFE1            jmp ecx
```

[그림 13] 문자열 복호화 분기점

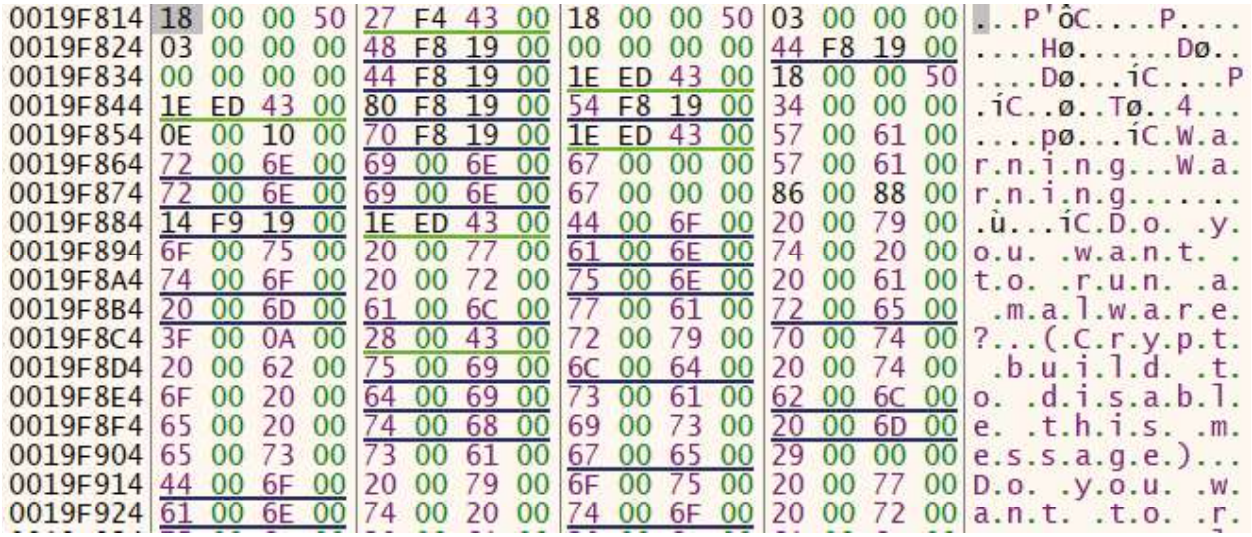
Lumma Stealer가 패킹이 되어있지 않으면 문자열을 복호화 하기 위한
ecx 레지스터에 저장된 주소로 jmp* 한다.

* jmp(어셈블리 명령어. 원하는 코드를 실행위치로 이동하는 명령어. 이하 jmp로 표기)



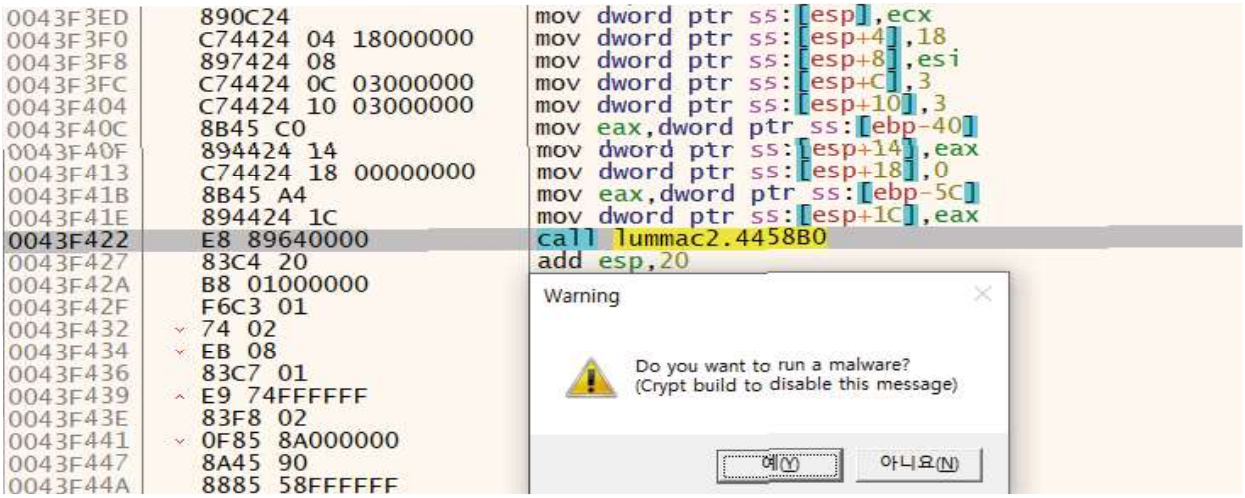
[그림 14] NtAllocateVirtualMemory

Heaven's gate 기법으로 NtAllocateVirtualMemory API를 호출하여 메모리를 할당한다.



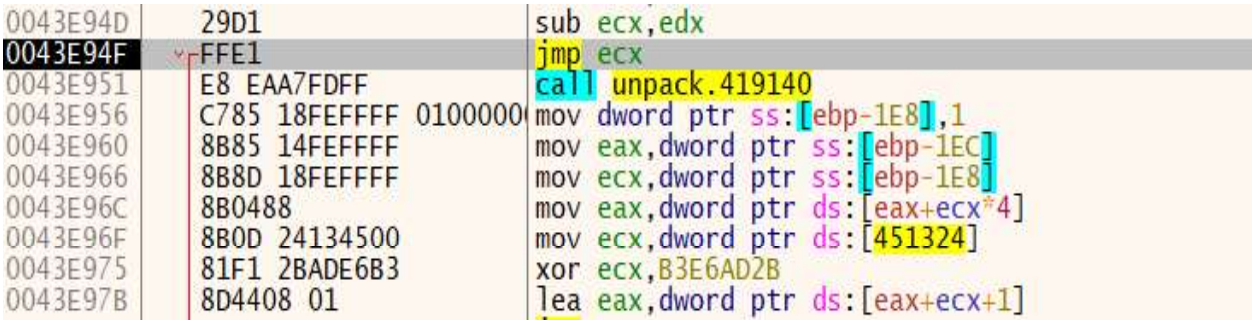
[그림 15] 문자열 난독화 해제

NtAllocateVirtualMemory로 할당한 메모리 영역에
난독화된 문자열을 복호화하여 메모리에 할당한다.



[그림 16] MessageBox

패킹이 되어있지 않은 Lumma Stealer 실행 시 멀웨어의 실행 여부를 알리는 메시지 박스를 띄운다.

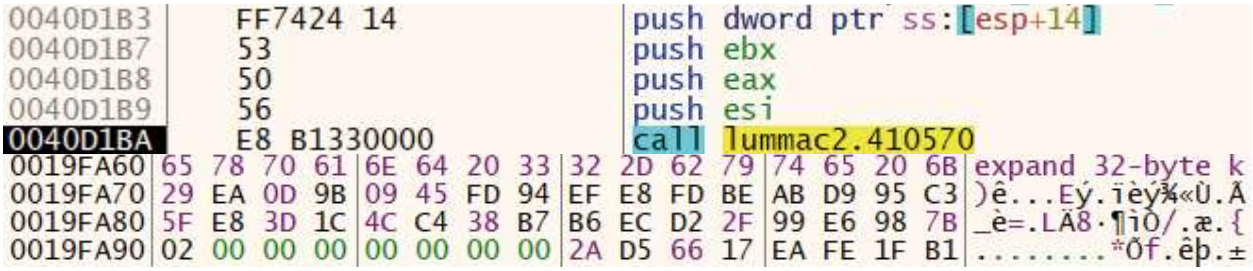


[그림 17] UPX 패킹된 샘플

Lumma Stealer가 패킹이 되어 있다면 문자열 복호화하는 주소가 아닌 곳으로 jmp 한다.
jmp 후 메시지 박스 출력 없이 진행하게 된다.

3.2 C2 서버 연결

Lumma Stealer 악성코드를 실행하면
Chrome, Firefox, Edge 등 주요 브라우저의
쿠키, 저장된 비밀번호, 자동완성 데이터, 신용카드 정보 등
개인 정보를 수집한다. 또한, 암호화폐 지갑도 탈취한다.

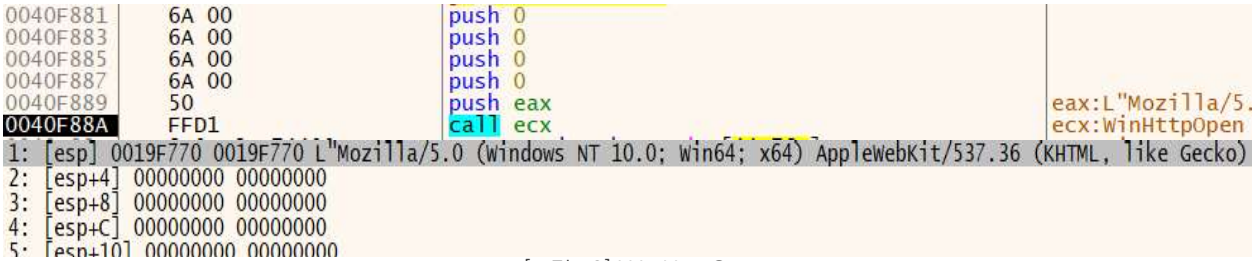


[그림 18] C2 도메인 복호화

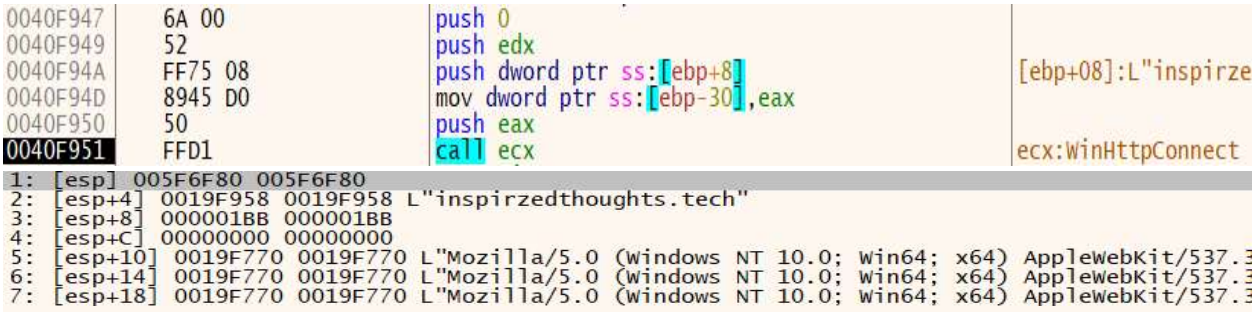
chacha20 암호 알고리즘으로 암호화되어 있는 C2 도메인 문자열을 복호화 한다.

복호화된 C2 도메인
· inspirzedthoughts.tech · pasteflawwed.world · hoyoverse.blog · dsfljsdfjewf.info · stomlegue.com · blast-hubs.com · blastikcn.com · decreaserid.world · lestagames.world

[표 3] C2 도메인

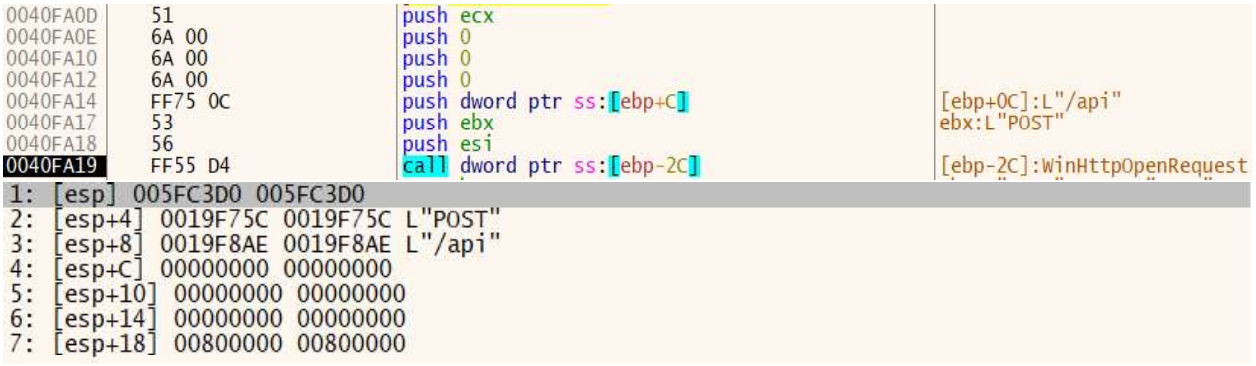


[그림 19] WinHttpOpen



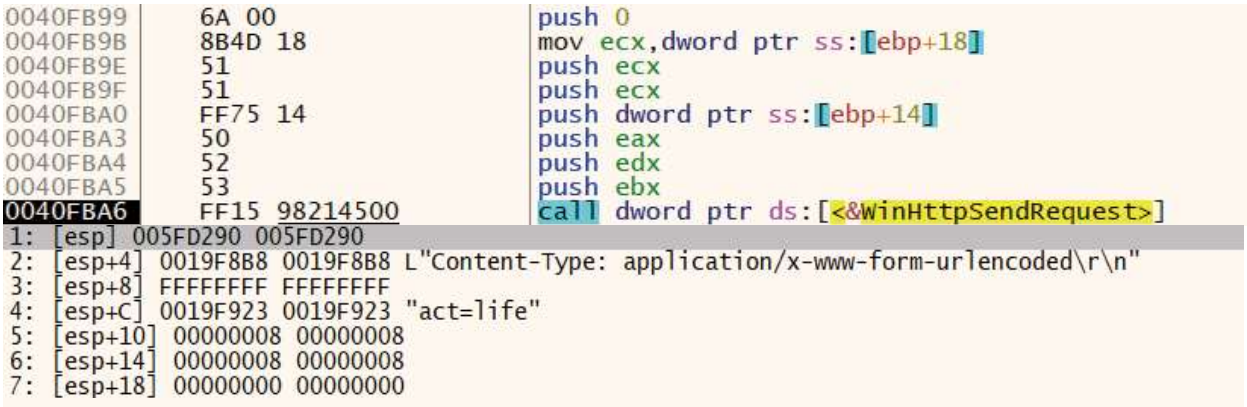
[그림 20] C2 연결

복호화한 C2 도메인을 WinHttpConnect API를 호출하여 443포트로 C2 연결을 시도한다.



[그림 21] WinHttpOpenRequest

inspirzedthoughts.tech C2에 POST 메서드로 /api 자원을 요청하는 핸들을 만든다.



[그림 22] WinHttpSendRequest

WinHttpSendRequest API를 호출하여
act=life라는 자원을 POST 요청으로 전송하여 C2 서버에 연결한다.
C2에 연결되면 C2 서버로부터 Base64로 인코딩된 config 파일을 받아온다.
해당 샘플의 C2 서버가 모두 닫혀있어 알려진 config 페이로드를 기반으로 작성하였다.

rLdnacjTGjrpxPtVh21J2bdzlGnt4d6gClEOT//KxanXlRFL8ul2GlrmwQ78Tz37jUO4S53D5IlvJH0qjbq3xsreCwzt8TYYh
ObBd61DPaHDUbhLl8PkgkM8fiCNvqTH2JchAKS2aWbGllk64QQlvJVftg3P2+zdJiosO93w9YWOx0VT6vZ7Spmgmi
HmSBWF9Rr6ClOCu4lmc2Ntxeik2dyaFB2noX8Ug7eUO6VBa6OVSbY+jl2yxX4iUmC9o6vlwtQCS+TxfhjT9YZ5/E89
+41DuEudw+SCLzB+P5ursciJ6zsrob17Vlqh2XmlAGvjlV3yAlOGu9InlXwmkb7rz9yVS0uy8SAYvqWXOelZOoWYMf0
HjI+9xSh9LCvd8PTUgMxFHerpKhbLtNlvpUgoqccX9R2MxIL8SDhgLpGpoIuAlQpL8vFpU4S0lzCqHj22xRLzDMOLrc
9kcyJthej/i/vWCwWtp2lmxoaSO+YDKryVX7YNz9vv3SYqLDvd8PWFjsdFU+r2e0qZoJoh5kgVhfIf8QqZk6vNVg15L
pOmoN3flUtLpfEgGMPm13f9T3P74BL4BYiVrfwlFGlqnL633MGVS0us8SALlOiAd/NPc+mbUeRL18P7wXohai6Lq+
D18PITAa2hf0+E5td36k9z+9wW7RqZjqzFKH0sNd3w5/7N2wsMvKBGFaywkzD1CDy0lV+2Dc/b790mKiw73fD1hY7
HRVPq9ntKmaCaleZlFYXyC/sNmJKC/G8pYSuKuevezdsLDLzxNhiE5sF3rU9l+81Rrku6gLLMbyV9E9CPvcblwhRL5P
F+GNP2hnn8Tz37jUO4S53D5IlvMH4/m6uxyInrOyWtt31fm+S3PPEla/WVHrZTz8v8jCgrLHXdnaTFwNlTGpT8Vl+No
54npyEgr9JRuEuJw+SSd311bYvo/5mAlRdL8vE/W5m0nzTzDGyF6xLgBoCIvfxWHWEsnqbl+tjYFQivtkZmhaGNMO
sJK/ubUfIL18P0giZzdG3F6JLIwNsCHbuPNXudq5Y85E9l+9NRrluQzaWCfnM0f9PotYuWlUIFp7B7Voi0izHmGSj86y/
XBoSPsc1jDVIMkKOrxsHeOzW/snZWjLCId6tPJPuNUb5LwcOkgjBzWS6TpqDd3+tIKqe6dFWErdl5pQlr44UOuBLPlf
yaOn0sP93w54zNxxcNqad7H7WYuiDzBTD58xbnApmOrvxWHWEsnqbl+tjYFQivtkZmhaGNMOsJK/ubUfIL18P0gi
ZzdG3F6JLIwNsCHbuPNXucsJMspyksqtwH+xnPzfzEKGs8MtOx592OjVdF6qM4AMvhmiX3CSit1IbINa+IqsNlOGAT
o72kxcDSExrq/zhXy/7Zf6VBa6OVSbY+jl2yxX4iUmC9o7HKw94JSau8aF/L6NkxpVd7pJslth3P2+6MKCEsdd3vpNnc
0wYdqfZGZoqrlnvrBCu8xQftR4eApthWDUchm6+9zMjzJUvk8XcY0+bRe+sIP7zbF/ZLwcOkgjBzWS6TpqDd3+tII4mL
QhqnoYx10Qg7qt4c+kvBw7qCMGNzY4TosYuWh0tLuPEgGMyliyXjDD24ki/ILIGEvdR4JGMTo72kxcDSExrq/zhXy/7Zf
6VBa6OVSbY+jl2yxX4iUmC6pqDK2MUSBOr/OF7L/skoqxZrrZVJpEXPkfyaKHRvP4+upN3Nkjs1ib1jfoy3kHerTyT7jVG
+R46OsMYofSw13fDn6NzHCwCrsm5ThqqICagsJ6DzFucCz838xChrPDLTsefdjo1XReqjOADL4Zol9wkordZWyDWr
iLLFUDhiI57o6YvBIV1LurZ5X4ewiDD1GyyrxF3sBIHD8oJwczRtvrq1xcXUBh2hvHRJteu9POsIE7DbH/VLwcO6gjBjc2
OE6LGLlodLS7jxIBjMpYsl4ww9uJlvyC+Ejbv6Yz1iLt3m58SOjUUaoad/V4iqmjLiH2eh2h+2Rc+b/JooEH4/k6OmyNjeC
Ae7jzV8gKieD+4BJbiVX7YNz9vs3SYqLDvd8PWFjsdFU+r2b0mMtosn6AsgtdJWtkXPjPyaKHsgJJ2uvYuAlR1L8vFbS
pmokjbmGSC22QDlRqaEu/Brln1t0+ihi5aFGkWz8W4Y0/TXd/dPc/uSA+YGipO/zWw4YiqM75n1/8MCCKXxNhiE5sF
39B4vt51RuEuXw+SCSyF+l5appN3F2AkalPxJToyllnerTy37jUHpRZbDqoIwYSJtj+j/i4nHFQavoXtXj62XMPRIFYXkB/
ElgL2Cw2U/aCaY6OmLwZVdS+LxNhiT5sF3xh05td4Q9R2EjrDTVn5dO5qrqPWD1AgHrrp9GMXmn3e9XzT1zFHgS9f
R8oJ6czRt2qu12cjWEwjtj0ZujKieMvUMJPnzFucCmY6ugiZzY23F6O/ajptFE+rpOHuZtJc85Aw9sNgd5zXCtbvMbzZ8L
pLo6YvIlV1btf9hGJ3mwWSrTyz7jSjvS4iP/JooNGQtnqanyMfYFwWru3ZdgaGYMeYBJLzSFvUDg4izyGdzlm2asOeTjvo
CHamee0mC5oZ5/E8st5VJtgeGg7flbDNrlJiroMjD0g8FrbV0UYagmTDhCjm+3B36S8HDu9ooaywCmr6k5M3EDEu1/2
EYjKrZb6UILrPeF/4LnYayyWk5aiyQpKHLz9UDB6q2f0qZo58l709l+9IJtlPPtazVeSUuGJ6mqczYlRpFs/F/VMv+2TzlAyy
z0xX+BICKrsNkPX4qnaGpxMLcCAynunJVj6GYd6tPLKOVSbY9n46w7GM/ZW2C5r6LydlFU+q1dFCBqZA97wAiu90Y/
wiPj7rDZD9hKJ6tqs7O2QIMq/E2GIy+2W+lPya33h20PoyNssV+c3NjhOigx46NRQ+gsXVSgKmaMOsPJrHHGfYLgYW
9zm0yYC6PpKHGyN0XS+Txf0DL/tkX7gMot9QWtCqFgLfOKgZvI5OvsYvRmxxLrb04AMuinTvlCCW/0xzxAIKJu85hO2
ctkaekzcbeAgGiu3tQi+bXd+lXa+OVKvsbgoj83SYqLCqR6P+LxtUDCKewdF+ArZI76gcrs9sU9QWFgrvMbTNrJZKkrM2
Om0UMsvEgGKetnSH+TzT1zFHxB8/b/MFsM20tlbqn2crWAwWjvnJXiqiTPuIIJr3aEvkKi429giZzazXd8OfkzcMPS7X/

[표 4-1] 인코딩된 config 파일

YRiMqtlvpQUnv9Yd/weCgrjFZTdsK5utpsDG2QoBprV0UY2qmjTjT2X70gm2U8+isclkM288mui4hdeVAgfq6T
hSmaOXM+8KL7ffEfMZh4W7zmA2YyuYpaDAzccXCK6/dBjF5p4vpVdrnsIS5g2Mw6OMcXNrId3w58vA2Q4M
obp8XIurkjnrBiez2RbkBoqLtsttP2Euj6umi4CVAhPq6Th/uJG6d/pBMvvSHbZTz4K0xWY3fiOPpqDLyN0NDKa/
dkqDp5k56gciv9QV8gelw/KCbyssdd2ApNHUlysAqrZoTpDmhnn8Tyy3lUm2D4SJtcFhN2EtlKeuw8PdFwqgo3xdiqm
TM+0DJLfRA/1LwcO72ihrLA2WvoTZ3JUaRbPxf1TL/tk+6Q4qsdMa8waPhrbFZDNsLpuuqsPG2QkIp7R8SpmikXerT
yyjlUm2ALqNqoJ3fXVtmqTnk47cDBmkv3FVja6eOegELbDSFvAGh465wWk3Zj+eo63GxJVLS62pOADLgZUe9RQ5r
ZUOuBLPhLCCMHNvIpSnr8PB2gEFrLd1XYSsiz/rAiC03gP2BouPuMpjOSxj3a+/i5aVMAalsXtOy7nXLqUIJ/uNUfoFj4y
wzmM7bSGTr6LCxt0XCq65eVaEp50y4Assv9MetkXPhKSCMHNDCqjqhvGOyksS6rZ0GNPmlTTpByS93B38AoSPt8
ZkOmkrnK2iysrZDw2psndXhK7ZeaUIM/uNUdMchI26gnd9dW2apOeTjtkMDay0dFmNrpw/4Q4tvdYe8g6OjLjOZjltLJ
GjqMDLIUtLrak4AMuXmiHyHyf7yl/vS4iP/JooMn4nl6aixMvWCgynt3RSgq6fOOwdKLfbFvgHgY62wWVzIm2asOeTj
vkCBoS6dF/LudcupQgn+41R+gGDirzLbTtnJJiprM7N0wgEo6NyW4WllTvjDiip3Ri2Rc+EpIIwc00jkLyg247KSxLqtnQY
0+aQJeECObLSH/kInYKxyXo0YymapKHEyNQAAaa2fVOEqtl5pQgz+41R2ACclL/MYyV3bYLmvovJ2UVT6rJzU4+ml
TfhAiup2hbxAoSRtsVvOGQmkq61x8DHABm48TYYjL7Zb6U5LKvFEscdjJW3z2Rzc2OE6KDHjo1FDaW4e1eKr5U64A
YuutMV/AGPhbrDbTlvLJehoMPJ1hdL5PF/QMv+2R7iHSWrlQ64Es+EsIIwc2gnkqygx8jQAwarvnlYhKKSPuMOJr7YFe
QBhIywy2Q+LGPdr7+LlpU0BqS/f07LudcupQgn+41R9weAgLPBazJmP4+krsPL2Q4FrKN+V4Klmj7iByex1ha2Rc+EpI
Iwc086jaXn1IDMRQym8SAYg6uRPeElJrzTGOQCio28xmM8aimeqLXNyt0GBqe/e1zL6Nkw/U9z++cc+BCAhK3IKCwi
NN2vq4uWlQEFuLp5U4ConjjgBSu00RL4AI6AtM9lPWYkIK+rz8mVS0utqTgAy424GskIMfvKX+9LiI/8mig/ZSGXo6DB
wNwLALije1yIr5ow7AErvNAf8QqEh7bBbnMibZqw55OO+QYLp6s4R8W/2TDpT3P73xDyBZ2IsMN6IX4rnKasx8nWC
wSpvX5Tg6eWO+wPLreVX7YMI8PkgkEZTW2C5r6LydlFU+q8dVGDqJk+7gQjsNkb5ASDi7jGaT5oJZCkp8/O0AYGq/
E2GIy+2W+lOSy0xx/xS5DNpYJvPyx13aerz8XRAg+jsnBdhqKQNu0GLLTeF/0IiIKxw2Q6ZCqX6OmLyc1FU+qHf0Crq
4N3+kEy+9IdtlPPjbHEaTJkJZ2urc/N3AYMo7dzW4GpnjHhDyC82xfzAIbD8oJvKyx13Y6E2dznCwix8WcWkuaeO6VX
a7DdHuQOhou4y2g3ZiCaqKLGy9ELD62xdFeMrpYz5QBr9ZUW7kvXw5zJfhJiJo/oulXXlQIH6uk4VoKnkTnpBy+p1Rr/
BI6MvMFpOWQ/m6isw8ndAQW4sHcYxeaeL6VXa4rDFvEEzaq72Wk5byaR6LiF15UCB+rpOFWHq50l6Q8rstIb5ASA
jr/CbSFtP5KjosjK2goHors4FsuhgXe9Twe4xBu0LJWVu855OGEh3bfp0o7SCUvy8XhZhrScNu8FJrPaFPMEi4e3zHo9Y
y2bo6bOzd4PBaOjOBbLoYF3vU8BoNQc+kmhiKrFKhJiJpqksYvRmxxLrb04AMuvlzvmCCOz0BH9C4CJssV6OWAnj6e
kxsrYDw6jo31RjebXd+IXa+OVKf0FvYCngnd9dW2apOeTjtYCCKu7cVSEoZ0l7wg5utQa+wePjrHIaTphIZCvoMTLlUtLrak
4AMuHlDzpVHH7yl/vS4iP/JooM2Yol6WkxM3HBA24sXVSmaySMugCJrjTF/0HnYq8wWNzIm2asOeTjvYSHaC2dE6Ap
5oh6B9rpJsltgyDw+SCbjpqKpumtc7I2goCo7VwW4uinTDgDCew0hL5D4aNtc0ofSwqhej/i+/OBgen8WcWkuaeO6VX
a7fbFPYBiYe5xGI2bCaep6PNytoFAKOwfI2BrZ865gkt+5tR8RPP2/zicz5gKt236dKO0glL8vFzVI+hmTrmBy6/3xT2A52L
vMV6lWwkmKSky8rTDA2ptTgWy6GBd71PBrfSOPEQz5zy2yg0YG3F6KbAxNoICKyyc12Bp54/6B0otNoV9gSJhb3Nbj
RlLJWv54WO0h1L8vFWX4ii2SirFmu82VGuS4qAu8RnO2opkqekwcTVDQyusHhdiKuXPe8MJ7XWHPgLz838xXBzNG
2+v7HB1ZUaRbPxf1TL/tk64gEjvdsX5AeAhbzDYj5gJpqmrM3L2AUNrLV8XIWI2XmlCDP7jVHRJr7Bn9V+OWshi6Omy
NjYFUu1/2EYjKrZb6UEJb7UHfwMgZG9yGQyayqWuqzZxd0GBaK4eFaLp5Q3pUFrvM1RrkuqoKvUYnFPOouioMfY3gQ
IvLxoGJTogHfiA2vjlRr5BYKIuMttO28omKKrx8/dDAGvtH5SiKiWNukLIrXcUbhLiJv8migFfCqFpbeJ/NYUGr+8aF7JiZo7
5gosq5UOuBLPhLCCMHNsKZGroMXB2AoMob5yVpmpnD/pByap3xvkAoaOssp6MCxj3a+/i5aVNwGpvW5VhOaGefx

[표 4-2] 인코딩된 config 파일

```
PLLeVSbYZnYO3wm89fiyVp63LyN4PCKO1dlGNp5Q25A8uu9wD+0vBw7vaKGssGpGjlsjYlRo05PF0GNOfgHf3T3P7k
h/7CoyNv9B6NW87nu+Z9evYCA6ktkZmqqyJOuoIFYXiAPEbzaW/1GtzIm2F6P+L798VBqW2OEfFv9klpVdr/Nsc9wiB
gK7QbjB6LtqWmerExQgEoaZ3ZrWRiDD1TQ24wxK2Rc+b/JooEmY9kKes3MGVGkWz8WoY0+beOegOKLXWA+QNj
JW/hVYNQyacuKraw9ETNZSWfI2MmKcA9Ag7+fMS4AjPzfzaKGssCputoIvRmxxLuPEgGMyoIDbmASipxxf1HYzEgvx
BPmMqi6uZ9fnEAhvol3tOiObXd/1Pc/v8HPkMmYD83SYqLD/d8OeMzccXDamnex+1mLQl4h8o+eQc8h2agKzFVg1BP
5q4pIn/wwYLpLY4Fsu+2W+IIjm8xRK0OpmAvMxvc3NjhOi1i5aVQgi4o35bnaXeCdsiObzFErQ6goeq12sjaxOjhbXM3tZ
HLpDzSU6IppcwpUFro5VJtiadhKzBKhZWb6y+pMvA0kUU5Kg4Ssv+2XDmHTm91gf1TLG9kdBvI29vrKWj3dvWFQyU
j1VKjLaadckIJrfrL8EaiJP+5GsIb23T6L+LIpUoGa2hexqnoZQ7pRBlopUDtIPPxLLPaTBiLo+6ocjY1kI1IJFoW5+hqDrhGT6
4xRbINa+Tv9ZvfEw9kL22zN7rOzy7tmgaraWPNKVBa6OVSbYrn4CoxSoCYSmLvaTbyZUaRbPxahjT5t456A4otdYD5A
2Mlb+FVg1NIJKlocbw6yMbqbR1VLWYribiH2md1gf1S8HDpIIwc00gkqWhxo7KSxLqozgAy+GXOuQMJbjHA/AImYD7/
FYSYSy8pab18PUVBr+gf0i1mK4m4h9pndYH9UvBw6SCMHNNIJyJqsqOyjoU5Kg4Tsv+yXmlHWvjlVb1GZ2Fv9RrdFIT
sqW/wMDbBjWUIXNIjKKULdsxGavYFf0FiJL8jCgrLHXdh6rTxdsLCOiVc0iMopQtpRAUpA
```

[표 4-3] 인코딩된 config 파일

인코딩된 문자열의 첫 32바이트는 복호화를 위한 키이고,
이후부터는 암호화된 본문이다. 32바이트 키와 XOR 연산하여 복호화를 진행하게 된다.
복호화된 내용은 [표 5]와 같다.

{“v”:1,”c”:[{“t”:0,”p”:”%userprofile%”,”m”:”*.txt”,”z”:”Important Files\/Profile”,”d”:2},
{“t”:0,”p”:”%appdata%\\Binance”,”m”:”app-store.json”,”z”:”Wallets\/Binance”,”d”:1},
{“t”:0,”p”:”%appdata%\\Binance”,”m”:”.finger-print.fp”,”z”:”Wallets\/Binance”,”d”:1},
{“t”:0,”p”:”%appdata%\\Binance”,”m”:”simple-storage.json”,”z”:”Wallets\/Binance”,”d”:1},
{“t”:0,”p”:”%appdata%\\Electrum\\wallets”,”m”:”*”,”z”:”Wallets\/Electrum”,”d”:1},
{“t”:0,”p”:”%appdata%\\Ethereum”,”m”:”keystore”,”z”:”Wallets\/Ethereum”,”d”:1},
{“t”:0,”p”:”%appdata%\\Exodus\\exodus.wallet”,”m”:”*”,”z”:”Wallets\/Exodus”,”d”:2},
{“t”:0,”p”:”%appdata%\\Ledger Live”,”m”:”*”,”z”:”Wallets\/Ledger Live”,”d”:2},
{“t”:0,”p”:”%appdata%\\atomic\\Local Storage\\leveldb”,”m”:”*”,”z”:”Wallets\/Atomic”,”d”:2},
{“t”:0,”p”:”%localappdata%\\Coinomi\\Coinomi\\wallets”,”m”:”*”,”z”:”Wallets\/Coinomi”,”d”:2},
{“t”:0,”p”:”%appdata%\\Authy Desktop\\Local Storage\\leveldb”,”m”:”*”,”z”:”Wallets\/Authy Desktop”,”d”:2},
{“t”:0,”p”:”%appdata%\\Bitcoin\\wallets”,”m”:”*”,”z”:”Wallets\/Bitcoin core”,”d”:2},
{“t”:0,”p”:”%appdata%\\com.liberty.jaxx\\IndexedDB”,”m”:”*.leveldb”,”z”:”Wallets\/JAXX New Version”,”d”:2},
{“t”:0,”p”:”%appdata%\\Electrum\\wallets”,”m”:”*”,”z”:”Wallets\/Electrum”,”d”:2},
{“t”:0,”p”:”%appdata%\\AnyDesk”,”m”:”*.conf”,”z”:”Applications\/AnyDesk”,”d”:2},
{“t”:0,”p”:”%appdata%\\FileZilla”,”m”:”recentservers.xml”,”z”:”Applications\/FileZilla”,”d”:2},
{“t”:0,”p”:”%appdata%\\FileZilla”,”m”:”sitemanager.xml”,”z”:”Applications\/FileZilla”,”d”:2},
{“t”:0,”p”:”%userprofile%”,”m”:”*.kbdx”,”z”:”Applications\/KeePass”,”d”:2},
{“t”:0,”p”:”%programfiles%\\Steam”,”m”:”ssfn*”,”z”:”Applications\/Steam”,”d”:2},
{“t”:0,”p”:”%programfiles%\\Steam\\config”,”m”:”*”,”z”:”Applications\/Steam\/config”,”d”:2},
{“t”:0,”p”:”%appdata%\\Telegram Desktop”,”m”:”*s”,”z”:”Applications\/Telegram”,”d”:2},
{“t”:1,”e”:[{“en”:”ejbalbakoplchlghecdalmeeeajnimhm”,”ez”:”MetaMask”},
{“en”:”nkbihfbeogaeaoehlefnkodbefgpgknn”,”ez”:”MetaMask”},
{“en”:”egjidjbpglichdcondbcbdnbeeppgdph”,”ez”:”Trust Wallet”},
{“en”:”ibnejdfjmmkpcnlpebklmnkoeoihofec”,”ez”:”TronLink”},
{“en”:”fnjhmkhhmkbjkkabndcnnogagogbneec”,”ez”:”Ronin Wallet”},
{“en”:”fhbohimaelbohpjbbldcngcnapndodjp”,”ez”:”Binance Chain Wallet”},
{“en”:”ffnbelfdoeiohenkjibnmadjiehjhajb”,”ez”:”Yoroi”},
{“en”:”jbdaocneiiinmjbjlgalhcelgbejmnid”,”ez”:”Nifty”},
{“en”:”afbcbjpbpfadlkmhmclhkeeodmamcflc”,”ez”:”Math”},
{“en”:”hnfanknocfeofbddgcijnmhnfnkdnaad”,”ez”:”Coinbase”},
{“en”:”hpglfhgfnhbgpjdenjgmdgoeiappafln”,”ez”:”Guarda”},
{“en”:”blnieiiffboillknjnepogjhkgnoapac”,”ez”:”EQUA”},
{“en”:”cjelfplplebdjjenllpjcblmjkfcffne”,”ez”:”Jaxx Liberty”},
{“en”:”fihkakfobkmkjojpchpfgcmhfjnmnfpi”,”ez”:”BitApp”},

[표 5-1] 복호화된 config 파일

{“en”:”kncchdigobghenbbaddojjnnaogfppfj”,”ez”:”iWlt”},
{“en”:”kkpllkodjeloidieedojogacfhpaihoh”,”ez”:”EnKrypt”},
{“en”:”amkmjjmmflddogmhpjloimipbofnfjih”,”ez”:”Wombat”},
{“en”:”nlbmnnijcnlegkjjpcfjclmcfggfefdm”,”ez”:”MEW CX”},
{“en”:”nanjmdknhkinifnkgdcggcfnhdaammmj”,”ez”:”Guild”},
{“en”:”nkddgncdjgjfcddamfgcmfnlhccnimig”,”ez”:”Saturn”},
{“en”:”cphhlgmgameodnhkjdmkpanlelnlohao”,”ez”:”NeoLine”},
{“en”:”nhnkbkgjikgcigadomkphalanndcapjk”,”ez”:”Clover”},
{“en”:”kpfopkelmapcoipemfendmdcghnegimn”,”ez”:”Liquality”},
{“en”:”aiifbnbfobpmeekipheeijimdpnlpgpp”,”ez”:”TerraStation”},
{“en”:”dmkamcknogkgcdfhhbddcghachkejeap”,”ez”:”Keplr”},
{“en”:”fhmfendgdocmcbmfikdcogofphimnkno”,”ez”:”Sollet”},
{“en”:”cnmamaachppnkjgnildpdmkaakejnhae”,”ez”:”Auro”},
{“en”:”jojhfeoedkpkglbfimdfabpdfjaoolaf”,”ez”:”Polymesh”},
{“en”:”flpiciilemghbmfalicajoolhkkenfe”,”ez”:”ICONex”},
{“en”:”nknhiehlklippafakaeklbeglecifhad”,”ez”:”Nabox”},
{“en”:”hcflpincpppdclinealmandijcmnkbgn”,”ez”:”KHC”},
{“en”:”ookjlbkiijinhpmnjffcofjonbfbgaoc”,”ez”:”Temple”},
{“en”:”mnfifefkajgofkcjkemidiaecocnkjeh”,”ez”:”TezBox”},
{“en”:”lodccjjbdhfakaekdiahmedfbieldgik”,”ez”:”DAppPlay”},
{“en”:”ijmpgkjfkbfhoebgogflfebnmejmfbm”,”ez”:”BitClip”},
{“en”:”lkcjlnjfpbikmcmbachjpdbijejflpcm”,”ez”:”Steem Keychain”},
{“en”:”onofpnbbkehpmmoabgpcpmigafmmnjh”,”ez”:”Nash Extension”},
{“en”:”bcopgchhojmggmffilplmbdicgaihlkp”,”ez”:”Hycon Lite Client”},
{“en”:”klnaejjgbibmhlephnhpmaofohgkpgkd”,”ez”:”ZilPay”},
{“en”:”aeachknmefphepccionboohckonoeemg”,”ez”:”Coin98”},
{“en”:”bhghoamapcdpbohphigoooaddinpkbai”,”ez”:”Authenticator”},
{“en”:”dkdedlpgdmmkkfjabffeganieamfklkm”,”ez”:”Cyano”},
{“en”:”nlgbhdfgdhgbiamfdfmbikcdghidoadd”,”ez”:”Byone”},
{“en”:”infeboajgfhgbjpjbeppbkgnabfdkdaf”,”ez”:”OneKey”},
{“en”:”cihmoadaighcejopammfbmddcmdekcje”,”ez”:”Leaf”},
{“en”:”gaedmjdfmmahhbjefcbgaolhhanlaolb”,”ez”:”Authy”},
{“en”:”oeljdldpnmdbchonielidgobddfffla”,”ez”:”EOS Authenticator”},
{“en”:”ilgcnhelpchnceeipipijaljkblbcob”,”ez”:”GAuth Authenticator”},
{“en”:”imloifkgjagghnncjkhggdhalmcnfklk”,”ez”:”Trezor Password Manager”},

[표 5-2] 복호화된 config 파일

{“en”:”bfnaelmomeimhlpmgjnjophhpkkoljpa”,”ez”:”Phantom”},
{“en”:”ppbibelpcjmhbdihakflkdcoccbgbkpo”,”ez”:”UniSat”}],”n”:[{“p”:”%localappdata%\\Google\\Chrome\\User Data”,”z”:”Chrome”},
{“p”:”%localappdata%\\Chromium\\User Data”,”z”:”Chromium”},
{“p”:”%localappdata%\\Microsoft\\Edge\\User Data”,”z”:”Edge”},
{“p”:”%localappdata%\\Kometa\\User Data”,”z”:”Kometa”},
{“p”:”%appdata%\\Opera Software\\Opera Stable”,”z”:”Opera Stable”},
{“p”:”%appdata%\\Opera Software\\Opera GX Stable”,”z”:”Opera GX Stable”},
{“p”:”%appdata%\\Opera Software\\Opera Neon\\User Data”,”z”:”Opera Neon”},
{“p”:”%localappdata%\\BraveSoftware\\Brave-Browser\\User Data”,”z”:”Brave Software”},
{“p”:”%localappdata%\\Comodo\\Dragon\\User Data”,”z”:”Comodo”},
{“p”:”%localappdata%\\CocCoc\\Browser\\User Data”,”z”:”CocCoc”}]},
{“t”:2,”p”:”%appdata%\\Mozilla\\Firefox\\Profiles”,”z”:”Mozilla Firefox”}]}

[표 5-3] 복호화된 config 파일

Lumma Stealer 4.0버전부터는
기존 Lumma Stealer와 다르게 정보 탈취 대상이 하드코딩되어 있지 않고 C2 서버로부터 받아온다.

```
0040FD8E   FF15 78214500   call dword ptr ds:[<&WinHttpCloseHandle>]
0040FD94   8B5D F0         mov ebx,dword ptr ss:[ebp-10]
0040FD97   FF75 DC         push dword ptr ss:[ebp-24]
0040FD9A   FF15 78214500   call dword ptr ds:[<&WinHttpCloseHandle>]
0040FDA0   FF75 D0         push dword ptr ss:[ebp-30]
0040FDA3   FF15 78214500   call dword ptr ds:[<&WinHttpCloseHandle>]
```

[그림 23] WinHttpCloseHandle

WinHttpOpenRequest로 생성한 자원을 요청하는 핸들을 종료시킨다.
만약 연결할 C2가 닫혀 있으면 다른 복호화된 C2 도메인에 [그림 19] ~ [그림 23] 과정을 반복한다.

3.3 추가 C2 서버 연결

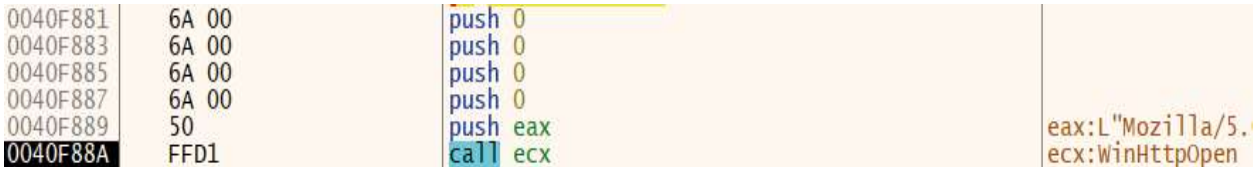
```
0040D970    0FB64C04 14          movzx ecx,byte ptr ss:[esp+eax+14]
0040D975    8D90 E1000000        lea edx,dword ptr ds:[eax+E1]
0040D97B    21CA                 and edx,ecx
0040D97D    89D6                 mov esi,edx
0040D97F    83E6 02              and esi,2
0040D982    89D7                 mov edi,edx
0040D984    83CA 02              or edx,2
0040D987    0FAFD6               imul edx,esi
0040D98A    83F6 02              xor esi,2
0040D98D    83E7 FD              and edi,FFFFFFFD
0040D990    0FAFFE               imul edi,esi
0040D993    01FA                 add edx,edi
0040D995    89D6                 mov esi,edx
0040D997    31CE                 xor esi,ecx
0040D999    F7D1                 not ecx
0040D99B    21D1                 and ecx,edx
0040D99D    8D1430               lea edx,dword ptr ds:[eax+esi]
0040D9A0    81C2 E1000000        add edx,E1
0040D9A6    01C9                 add ecx,ecx
0040D9A8    29CA                 sub edx,ecx
0040D9AA    80C2 A6              add dl,A6
0040D9AD    885404 14            mov byte ptr ss:[esp+eax+14],dl
0040D9B1    89C1                 mov ecx,eax
0040D9B3    83E1 01              and ecx,1
0040D9B6    83F0 01              xor eax,1
0040D9B9    8D0448               lea eax,dword ptr ds:[eax+ecx*2]
0040D9BC    83F8 6C              cmp eax,6C
0040D9BF  ^ 72 AF                jb lummac2.40D970
1: [esp] 0019F958 0019F958 L"https://steamcommunity.com/profiles/76561199822375128"
2: [esp+4] 0019F94C 0019F94C
3: [esp+8] 0019F950 0019F950
4: [esp+C] 00000000 00000000
5: [esp+10] 00000000 00000000
6: [esp+14] 0019F95A 0019F95A L"ttps://steamcommunity.com/profiles/76561199822375128"
7: [esp+18] 00000000 00000000
```

[그림 24] 추가 C2 도메인 복호화

Lumma Stealer는 Steamcommunity 프로필도 C2로 사용한다.
[표 3]의 C2 서버가 모두 닫혀있으면
Steamcommunity URL을 복호화하여 WinHttp API로 연결을 시도한다.

3.4 정보 탈취

복호화된 config 파일에 명시된
브라우저 쿠키 및 저장된 개인 정보와 암호화폐 지갑 주소를 탈취하여 C2 서버에 전송한다.



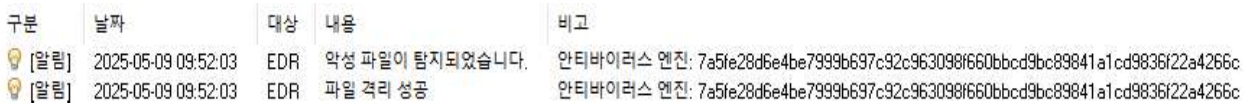
[그림 25] WinHttpOpen

WinHttpOpen API를 호출하여 C2 연결을 위해
WinHTTP 함수를 초기화하고 WinHTTP 세션 핸들을 반환한다.

[그림 26] C2 연결 정보

이후 탈취한 데이터를 “act=receive_message&ver=4.0&lid=yau6Na–5504031492&j=” 형식으로
POST 요청으로 데이터를 C2 서버로 보낸다.
lid는 해당 Lumma Stealer의 식별 번호로 추정된다.

4. 탐지



구분	날짜	대상	내용	비고
[알림]	2025-05-09 09:52:03	EDR	악성 파일이 탐지되었습니다.	안티바이러스 엔진: 7a5fe28d6e4be7999b697c92c963098f660bbcd9bc89841a1cd9836f22a4266c
[알림]	2025-05-09 09:52:03	EDR	파일 격리 성공	안티바이러스 엔진: 7a5fe28d6e4be7999b697c92c963098f660bbcd9bc89841a1cd9836f22a4266c

[그림 27] Privacy-i EDR 안티바이러스 탐지

Lumma Stealer는 Privacy-i EDR의 안티바이러스 엔진에 의해 탐지 및 격리된다.

5. 대응

1. MS에서 제공하는 보안 업데이트를 자동으로 설정한다.
2. 사용 중인 소프트웨어의 최신 업데이트 상태를 유지한다.
3. 백신 최신 업데이트 상태를 유지한다.
4. 주요 문서는 주기적으로 백업하고 물리적으로 분리하여 관리한다.
5. 신뢰할 수 없는 메일의 첨부파일은 실행을 금지한다.
6. 비 업무 사이트 및 신뢰할 수 없는 웹사이트의 연결을 차단한다.

궁금하신 점이나 문의사항은 malware@somansa.com 으로 문의주십시오.