SOMANSA

MALWARE ANALYSIS REPORT

No.66 | 2025 년 07 월

A 신용보증기업

랜섬웨어 감염으로 전산서비스 전면 중단

전세·사업대금 보증 지연원인

GUNRA 랜섬웨어

(주) 소만사 악성코드 분석 센터

목 차

1. 개 요

A. 개요

• 2025 년 7 월 14 일 00 시 40 분, A 신용보증기업에서 랜섬웨어 감염이 최초로 감지됨.

• 2025 년 7 월 14 일 05 시 05 분, A 신용보증기업은 금융보안원, KISA 등에 지원을 요청함.

• 공격에 사용된 'Gunra' 랜섬웨어는 2025 년 4 월에 처음 등장하였으며,

기존 콘티(Conti) v2 의 소스코드를 기반으로 개발된 고도화 된 변종

• 해당 랜섬웨어 샘플은 2025 년 7 월 12 일 제작되었고,

2025 년 7 월 14 일 VirusTotal 에서 최초로 발견됨.

(아래 분석 샘플은 실제 A 신용보증기업 공격에 사용된 파일은 아니나

동일 변종으로 유사한 악성행위를 수행함.)

• 보안 제품의 탐지를 우회하기 위해 백신 프로세스를 강제로 종료하며,

분석 환경에서의 실행을 회피하는 기능을 보유함.

•피해자의 파일 복구를 방해하기 위해 볼륨 섀도 복사본(Volume Shadow Copy)을 삭제함.

2. 정 보

2.1 침해 지표

PE 실행파일 (SHA-256)

- 91f8fc7a3290611e28a35a403fd815554d9d856006cc2ee91ccdb64057ae53b0

2.2 MITRE ATT&CK

Discovery

- **(T1082)** System Information Discovery

Collection

- **(T1083)** File and Directory Discovery

Impact

- **(T1486)** Data Encrypted for Impact
- **(T1490)** Inhibit System Recovery

3. 분 석

3.1 쉐도우 복사본 삭제 명령 수행

Windows Management Instrumentation(WMI) 유틸리티를 사용하여

사용 가능한 섀도 복사본을 삭제 시도한다.

```
        0LL);
sub_7FF6E2971773((__int64)v35, 2048LL);
v12 = sub_7FF6E2971813(
        v42,
        L"cmd.exe /c C:\\Windows\\System32\\wbem\\WMIC.exe shadowcopy where \"ID='%s'\" delete");
v13 = (const WCHAR *)sub_7FF6E2972088(v12);
wsprintfW(v35, v13, v34);
sub_7FF6E2972AE7(v36);
sub_7FF6E2972B78(v35);
```

```
00007FF967F0CEA7    48:8B8424 A8000000      mov rax,qword ptr ss:[rsp+A8]
00007FF967F0CEAF    49:8943 F0              mov qword ptr ds:[r11-10],rax
00007FF967F0CEB3    48:8B8424 A0000000      mov rax,qword ptr ss:[rsp+A0]
00007FF967F0CEBB    49:8943 E8              mov qword ptr ds:[r11-18],rax
00007FF967F0CEBF    48:8B8424 98000000      mov rax,qword ptr ss:[rsp+98]
00007FF967F0CEC7    49:8943 E0              mov qword ptr ds:[r11-20],rax
00007FF967F0CECB    48:8B8424 90000000      mov rax,qword ptr ss:[rsp+90]
00007FF967F0CED3    49:8943 D8              mov qword ptr ds:[r11-28],rax
00007FF967F0CED7    8B8424 88000000         mov eax,dword ptr ss:[rsp+88]
00007FF967F0CEDE    894424 28               mov dword ptr ss:[rsp+28],eax
00007FF967F0CEE2    8B8424 80000000         mov eax,dword ptr ss:[rsp+80]
00007FF967F0CEE9    894424 20               mov dword ptr ss:[rsp+20],eax
00007FF967F0CEED    48:FF15 4C800600        call qword ptr ds:[<CreateProcessW>]
```

```
000000000014DD30  63 00 6D 00 64 00 2E 00 65 00 78 00 65 00 20 00  c.m.d...e.x.e. .
000000000014DD40  2F 00 63 00 20 00 43 00 3A 00 5C 00 57 00 69 00  /.c. .C.:.\.W.i.
000000000014DD50  6E 00 64 00 6F 00 77 00 73 00 5C 00 53 00 79 00  n.d.o.w.s.\.S.y.
000000000014DD60  73 00 74 00 65 00 6D 00 33 00 32 00 5C 00 77 00  s.t.e.m.3.2.\.w.
000000000014DD70  62 00 65 00 6D 00 5C 00 57 00 4D 00 49 00 43 00  b.e.m.\.W.M.I.C.
000000000014DD80  2E 00 65 00 78 00 65 00 20 00 73 00 68 00 61 00  ..e.x.e. .s.h.a.
000000000014DD90  64 00 6F 00 77 00 63 00 6F 00 70 00 79 00 20 00  d.o.w.c.o.p.y. .
000000000014DDA0  77 00 68 00 65 00 72 00 65 00 20 00 22 00 49 00  w.h.e.r.e. .".I.
000000000014DDB0  44 00 3D 00 27 00 7B 00 31 00 31 00 42 00 30 00  D.=.'.{.1.1.B.0.
000000000014DDC0  36 00 41 00 44 00 44 00 2D 00 33 00 44 00 45 00  6.A.D.D.-.3.D.E.
000000000014DDD0  36 00 2D 00 34 00 43 00 37 00 36 00 2D 00 39 00  6.-.4.C.7.6.-.9.
000000000014DDE0  35 00 44 00 33 00 2D 00 32 00 34 00 31 00 38 00  5.D.3.-.2.4.1.8.
000000000014DDF0  44 00 31 00 33 00 34 00 34 00 39 00 30 00 33 00  D.1.3.4.4.9.0.3.
000000000014DE00  7D 00 27 00 22 00 20 00 64 00 65 00 6C 00 65 00  }.'.". .d.e.l.e.
000000000014DE10  74 00 65 00 00 00 00 00 00 00 00 00 00 00 00 00  t.e.............
```

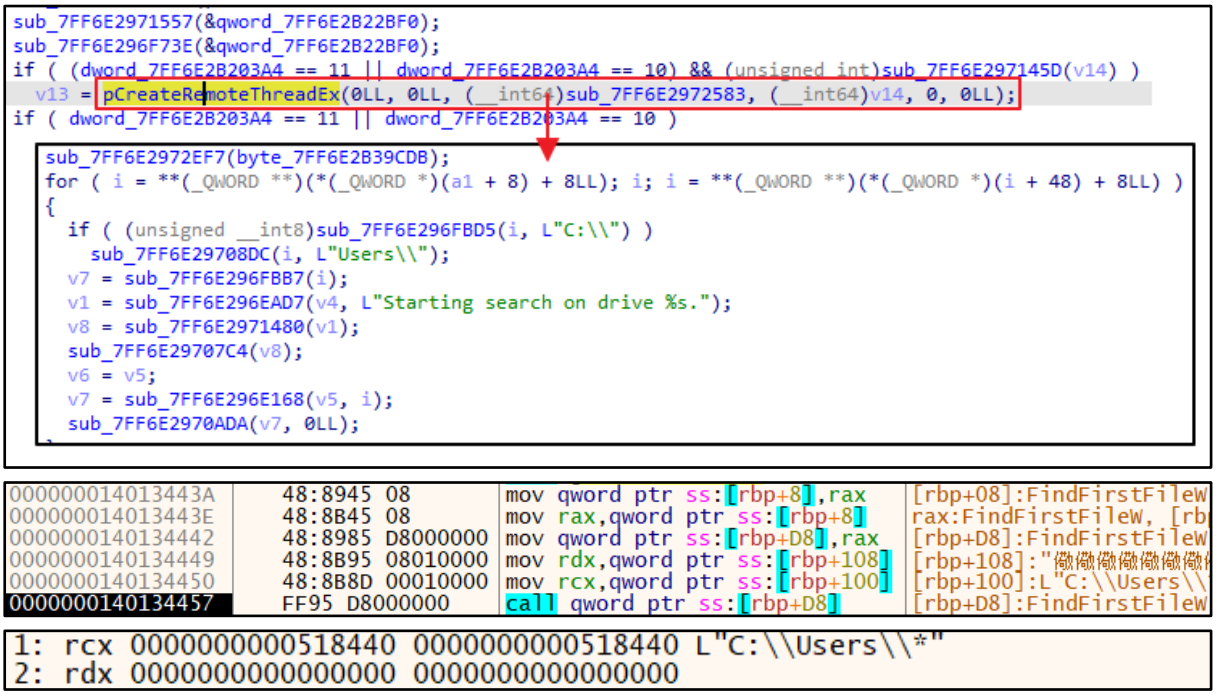
[그림 1] CMD 프로세스 생성 후 WMIC.exe 를 실행시켜 shadowcopy 삭제 시도

3.2 파일 암호화

Gunra 랜섬웨어는 파일 암호화를 위해 다음과 같은 절차를 수행한다.

1. FindFirstFileW 함수를 이용하여 대상 파일을 탐색한 뒤,
   해당 파일이 암호화 예외 디렉터리에 존재하는지 확인한다.
2. 파일의 확장자를 검사하여 암호화 예외 대상인지 여부를 판단한다.
3. 암호화 조건을 만족한 경우, ChaCha 및 RSA 알고리즘 기반의 암호화 키를 생성한다.
4. 파일의 속성을 조회한 뒤,
   암호화 대상 확장자에 해당하면 실제 파일 암호화를 수행한다.

“D:\”와 “C:\Users\” 디렉터리 하위의 모든 파일을 대상으로 암호화 탐색을 수행한다.



[그림 2] 암호화 대상 디렉터리 탐색

대상 파일의 속성을 확인하여 숨김 파일인지, 읽기 전용인지 등을 확인한다.

```
00000001401193E5    48:8945 08          mov qword ptr ss:[rbp+8],rax          [rbp+08]:GetFileAttributesW
00000001401193E9    48:8B45 08          mov rax,qword ptr ss:[rbp+8]          rax:GetFileAttributesW, [rb
00000001401193ED    48:8985 D8000000    mov qword ptr ss:[rbp+D8],rax         [rbp+D8]:GetFileAttributesW
00000001401193F4    48:8B8D 00010000    mov rcx,qword ptr ss:[rbp+100]        [rbp+100]:L"C:\\Users\\PC\\
00000001401193FB    FF95 D8000000       call qword ptr ss:[rbp+D8]            [rbp+D8]:GetFileAttributesW

1: rcx 00000000004FDF50 00000000004FDF50 L"C:\\Users\\PC\\Desktop\\AfMUaI.pdf"
2: rdx 0000000000000000 0000000000000000
3: r8 0000000093AFB23A 0000000093AFB23A
4: r9 000000000000000D 000000000000000D
5: [rsp+20] 0000000000000200 0000000000000200
```

[그림 3] 파일 속성 확인

디렉터리에 [그림 4]와 일치하는 문자열이 있으면 암호화 대상 경로에서 제외한다.

```
sub_1400B2EF7((__int64)&unk_140279CDB);
v3 = sub_1400B2056(v21, L"tmp");
v18[0] = sub_1400AFDF6(v3);
v4 = sub_1400B091D(v22, L"winnt");
v18[1] = sub_1400AE154(v4);
v5 = sub_1400AE3CF(v23, L"temp");
v18[2] = sub_1400AF7B1(v5);
v6 = sub_1400AE947(v24, L"thumb");
v18[3] = sub_1400AFCA2(v6);
v7 = sub_1400AF603(v25, L"$Recycle.Bin");
v18[4] = sub_1400AE799(v7);
v8 = sub_1400B1908(v26, L"$RECYCLE.BIN");
v18[5] = sub_1400AE4BF(v8);
v9 = sub_1400AE514(v27, L"System Volume Information");
v18[6] = sub_1400AF833(v9);
v10 = sub_1400AF1FD(v28, L"Boot");
v18[7] = sub_1400B238A(v10);
v11 = sub_1400B1877(v29, L"Windows");
v18[8] = sub_1400B16A1(v11);
v12 = sub_1400B11A1(v30, L"Trend Micro");
v18[9] = sub_1400B339D(v12);
v19 = 10;
```

[그림 4] 암호화 예외 디렉터리

tmp	winnt	temp	thumb
$Recycle.Bin	$RECYCLE.BIN	System Volume Information	Boot
Windows	Trend Micro		

[표 2] 암호화 예외 디렉터리 리스트

```
sub_1400B2EF7((__int64)&unk_140279CDB);
v3 = sub_1400AF545(v19, L".exe");
v16[0] = sub_1400AFE50(v3);
v4 = sub_1400B0CEC(v20, L".dll");
v16[1] = sub_1400B0CD8(v4);
v5 = sub_1400AF67B(v21, L".lnk");
v16[2] = sub_1400B04A4(v5);
v6 = sub_1400AF685(v22, L".sys");
v16[3] = sub_1400B13D6(v6);
v7 = sub_1400B3258(v23, L".msi");
v16[4] = sub_1400B1296(v7);
v8 = sub_1400B2F5B(v24, L"R3ADM3.txt");
v16[5] = sub_1400B22AE(v8);
v9 = sub_1400AFAD6(v25, L"CONTI_LOG.txt");
v16[6] = sub_1400B1B15(v9);
```

[그림 5] 암호화 예외 확장자

exe	dll	lnk	sys
msi	R3ADM3.txt	CONTI_LOG.txt	

[표 3] 암호화 예외 확장자 리스트

```
0000000140119C4D   48:8945 08       mov qword ptr ss:[rbp+8],rax      [rbp+08]:StrStrIW,
0000000140119C51   48:8B45 08       mov rax,qword ptr ss:[rbp+8]      rax:StrStrIW, [rbp
0000000140119C55   48:8985 D8000000 mov qword ptr ss:[rbp+D8],rax     [rbp+D8]:StrStrIW,
0000000140119C5C   48:8B95 08010000 mov rdx,qword ptr ss:[rbp+108]    [rbp+108]:L".4dd"
0000000140119C63   48:8B8D 00010000 mov rcx,qword ptr ss:[rbp+100]    [rbp+100]:L"C:\\Us
0000000140119C6A   FF95 D8000000    call qword ptr ss:[rbp+D8]        [rbp+D8]:StrStrIW
```

```
1: rcx 000000000052D220 000000000052D220 L"C:\\Users\\PC\\Desktop\\AfMUaI.pdf"
2: rdx 00000000021AD7A9 00000000021AD7A9 L".4dd"
3: r8 000000005A8CE5B8 000000005A8CE5B8
4: r9 000000000000004A 000000000000004A
5: [rsp+20] 0000000000000000 0000000000000000
6: [rsp+28] 00007FF9666D6AF0 <shlwapi.StrStrIW> (00007FF9666D6AF0)
```

[그림 6] 암호화 대상 확장자 비교

코드 내부에 하드코딩된 형태로 다음과 같은 대상 확장자 목록이 저장되어 있다.

4dd	4dl	accdb	accdc	accde	accdr	accdt	accft
adb	ade	adf	adp	arc	ora	alf	ask
btr	bdf	cat	cdb	ckp	cma	cdp	dacpac
dad	dadiagrams	daschema	db	db-shm	db-wal	db3	dbc
dbf	dbs	dbt	dbv	dbx	dcb	dct	dcx
ddl	dlis	dp1	dqy	dsk	dsn dtsx	dxl	eco
ecx	edb	epim	exb	fcd	fdb	fic	fmp
fmp12	fmpsl	fol	fp3	fp4	fp5	fp7	fpt
frm	gdb	grdb	gwi	hdb	his	ib	idb
ihx	itdb	itw	jet	jtx	kdb	kexi	kexic
kexis	lgc	lwx	maf	maq	mar	mas	mav
mdb	mdf	mpd	mrg	mud	mwb	myd	ndf
nnt	nrmlib	ns2	ns3	ns4	nsf	nv	nv2
nwdb	nyf	odb	oqy	orx	owc	qry	qvd
rbf	rctd	rod	p96	p97	pan	pdb	pdm
pnz	rodx	rpd	rsd	sas7bdat	sbf	scx	sdb
sdc	sdf	sis	spq	sql	sqlite	sqlite3	sqlitedb
te	temx	tmd	tps	trc	trm	udb	udl
usr	v12	vis	vpd	vvv	wdb	wmdb	xdb
xld	xmlff	abcddb	abs	accdw	adn	db2	fm5 hjt
icg	icr	kdb	lut	maw	mdn	mdt	vdi
vhd	vmdk	pvm	vmem	vmsn	vmsd	nvram	vmx
raw	qcow2	subvol	bin	vsv	avhd	vmrs	vhdx
avdx	vmcx	iso					

[표 4] 암호화 대상 확장자

파일을 암호화한 후 파일 확장자(이름)뒤에 “.CRYPT”를 붙인다.

```
00007FF967F12BD0 <ker    48:83EC 38              sub rsp,38
00007FF967F12BD4         45:33C9                 xor r9d,r9d
00007FF967F12BD7         C74424 20 02000000      mov dword ptr ss:[rsp+20],2
00007FF967F12BDF         45:33C0                 xor r8d,r8d
00007FF967F12BE2         48:FF15 171B0600        call qword ptr ds:[<MoveFileWithProgressW>]

1: rcx 00000000004FDE20 00000000004FDE20 L"C:\\Users\\PC\\Desktop\\n7Wpa.docx"
2: rdx 0000000000505F70 0000000000505F70 L"C:\\Users\\PC\\Desktop\\n7Wpa.docx.CRYPT"
```

[그림 7] 암호화 파일 확장자 추가

3.3 암호화 알고리즘

Gunra 랜섬웨어는 파일 암호화를 위해 ChaCha 과 RSA 알고리즘을 사용한다.

```
00000001401192F1    48:8945 08          mov qword ptr ss:[rbp+8],rax          [rbp+08]:CryptGenRandom,
00000001401192F5    48:8B45 08          mov rax,qword ptr ss:[rbp+8]          rax:CryptGenRandom, [rbp
00000001401192F9    48:8985 D8000000    mov qword ptr ss:[rbp+D8],rax         [rbp+D8]:CryptGenRandom,
0000000140119300    4C:8B85 10010000    mov r8,qword ptr ss:[rbp+110]
0000000140119307    8B95 08010000       mov edx,dword ptr ss:[rbp+108]
000000014011930D    48:8B8D 00010000    mov rcx,qword ptr ss:[rbp+100]
0000000140119314    FF95 D8000000       call qword ptr ss:[rbp+D8]            [rbp+D8]:CryptGenRandom

00000000021AFAD8  38 61 70 68 29 6E 37 86  33 DA 65 BE 84 8F B9 9A  8aph)n7.3Úe¾..¹.
00000000021AFAE8  AB FA A4 98 EB BC 74 B3  62 FA 3E 20 35 E1 DA 92  «ú¤.ë¼t³bú> 5áÚ.
00000000021AFAF8  0D 03 FA 9B 3A 0D BD DE  00 00 00 00 00 00 00 00  ..ú.:.½Þ........
```

[그림 8] RSA 키 생성

```
a1[4] = *a2;
a1[5] = a2[1];
a1[6] = a2[2];
a1[7] = a2[3];
if ( a3 == 256 )
{
  a2 += 4;
  v4 = "expand 32-byte kexpand 16-byte k";
}
else
{
  v4 = "expand 16-byte k";
}
a1[8] = *a2;
a1[9] = a2[1];
a1[10] = a2[2];
a1[11] = a2[3];
*a1 = *(_DWORD *)v4;
a1[1] = *((_DWORD *)v4 + 1);
a1[2] = *((_DWORD *)v4 + 2);
result = *((unsigned int *)v4 + 3);
a1[3] = result;
return result;
```

```
a1[12] = 0;
a1[13] = 0;
a1[14] = *a2;
result = (unsigned int)a2[1];
a1[15] = result;
return result;
```

```
00000000021AFA98 65 78 70 61 6E 64 20 33 32 2D 62 79 74 65 20 6B expand 32-byte k
00000000021AFAA8 33 DA 65 BE 84 8F B9 9A AB FA A4 98 EB BC 74 B3 3Úe¾..¹.«ú¤.ë¼t³
00000000021AFAB8 62 FA 3E 20 35 E1 DA 92 0D 03 FA 9B 3A 0D BD DE bú> 5áÚ...ú.:.½Þ
00000000021AFAC8 00 00 00 00 00 00 00 00 38 61 70 68 29 6E 37 86 ........8aph)n7.
```

[그림 9] ChaCha 키 생성

4. 탐 지

구분	날짜	대상	내용	비고
[알림]	2025-07-17 20:19:51	EDR	이상 파일 I/O 행위에 의해 손상된 파일의 복구...	Ransomware: Gunra.exe
[알림]	2025-07-17 20:19:51	EDR	이상 파일 I/O 행위에 의해 손상된 파일 복구 결과	프로세스: \??\C:\Users\PC\Desktop\Gunra.exe, 전체: 14, 성공: 14, 실패: 0
[알림]	2025-07-17 20:19:51	EDR	이상 파일 I/O 행위에 의해 손상된 파일 복구 성공	C:\Users\PC\Desktop\die_win64_portable_3.09_x64.zip
[알림]	2025-07-17 20:19:51	EDR	이상 파일 I/O 행위에 의해 손상된 파일 복구 성공	C:\Users\PC\Desktop\dnSpy-net-win32.zip
[알림]	2025-07-17 20:19:51	EDR	이상 파일 I/O 행위에 의해 손상된 파일 복구 성공	C:\Users\PC\Desktop\dnSpy-net-win64.zip

에이전트 IP	사용자	위험도	분류	이상 행위	파일 이름	대응 결과	MITRE ATT&CK
192.168.64.132	ktw(ktw)	높음	악성코드	랜섬웨어 / Ransomware	Gunra.exe		Data Encrypted for Impact Inhibit System Recovery Unsecured Credentials Credentials In Files
192.168.64.132	ktw(ktw)	중간	익스플로잇	디스크 볼륨 백업본 삭제 / impact.impair.volume-shadowcopy	cmd.exe		Inhibit System Recovery

[그림 10] Privacy-i EDR 탐지

Privacy-i EDR 의 행위 기반 탐지 엔진은

쉐도우 복사본 삭제 및 랜섬웨어의 파일 암호화와 같이 시스템 내에서 발생하는

비정상적인 I/O 를 탐지하여

Gunra 랜섬웨어를 차단하고 감염된 파일을 실시간 복구하였다.

5. 대 응

1) Privacy-i EDR 과 같은 EDR 제품을 통해 취약점 실행을 행위 기반으로 차단한다.

2) 볼륨 섀도 복사본 활용방식이 아닌 실시간 백업/복구 기능으로 데이터를 보호한다.

3) 논리적 망분리를 적용하여 악성코드 유입을 원천 차단한다.

4) AV(패턴기반탐지) + EDR(행위기반탐지) 솔루션으로 신변종 악성코드 랜섬웨어 대응한다.

5) 서버 취약점을 주기적으로 점검 및 보완 한다.

궁금하신 점이나 문의사항은 malware@somansa.com 으로 전달 부탁드립니다.