

# ThunderCats Hack the FSB | Your Taxes Didn't Pay For This Op - SentinelLabs

By Juan Andrés Guerrero-Saade

Published: 2021-06-08 · Archived: 2026-04-05 20:27:36 UTC

## Key Findings

- This research focuses on the 'Mail-O' malware used against the FSB and other Russian government organizations, detailed in the May 2021 FSB NKTsKI and Rostelecom-Solar report.
- Early armchair commentary presumed that given the targets, this attack would undoubtedly be the work of a Western government, Five Eyes, or the United States.
- Our analysis disproves that hypothesis.
- Instead, we present the argument that the Mail-O malware is a variant of a relatively well-known malware called PhantomNet or SManager used by a threat actor 'TA428'
- Previous reporting on TA428 points to Chinese origin and details a history of attacks against South East Asian and Russian targets.

## Actor Disambiguation

**Related actors:** TA428, suspected IronHusky

**Related operations:** Operation SignSight, Operation LagTimeIT

**Related malware:** PhantomNet, SManager, TManger, CoughingDown

In May 2021, the Russian Federal Security Service's National Coordination Center for Computer Incidents (NKTsKI) in coordination with Rostelecom [announced](#) that several Russian government institutions had been victims of an APT campaign. While the Russian government has [made a similar announcement before](#), it's the first time they've accompanied it with a moderately detailed technical analysis. Several researchers, myself included, jumped on the opportunity to write our YARA rules and hope for a glimpse at the culprit.

The InfoSec twitterverse needed no such artifacts as blind speculation immediately pointed at a Western government, Five Eyes, or the United States as *de facto* culprits. I think we'll be relieved to find out that was most likely not the case – if solely because we've come to expect a higher standard for Western malware development.

Initial attempts to find the samples were fruitless but that changed this past weekend as some kind soul (or more likely a bulk autosubmitter) uploaded a copy of the 'Mail-O' malware to [VirusTotal](#). We track this activity under the name 'ThunderCats'.

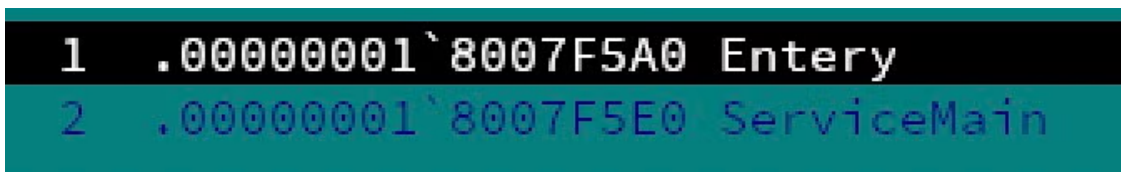
## Technical Analysis

```
SHA256
603881f4c80e9910ab22f39717e8b296910bff08cd0f25f78d5bff1ae0dce5d7
SHA1
b7c1ec9484c4c2dcd01f861eeaa3b915c3e3312e
MD5
d58b95f8413f784552d7fdadbb621243
Size
2.82 MB
Compilation Timestamp
2019-12-20 02:13:01
First Submitted
2021-06-05 05:22:04
```

In line with the findings of the NKTsKI-Rostelecom report, the Mail-O malware acts as a downloader with a thin veneer of similarity to the legitimate [Mail.ru Disk-O software](#). The disguise consists of a version number (“19.05.0045”) lifted from a legitimate Disk-O executable and the use of a real Mail.ru to post victim details and host a next stage payload.

The executable is bulked up to 2.8MB by statically linking both libcurl 7.64.1 and OpenSSL. Focus becomes important to avoid going down a pointless rabbit hole of reversing unrelated open-source code. For that reason, we should focus primarily on the exported functions.

The Mail-O malware exports two functions, `Entery` and `ServiceMain` :



```
1 .00000001`8007F5A0 Entery
2 .00000001`8007F5E0 ServiceMain
```

Mail-O malware’s exported functions

## Mail-O: ServiceMain

```
void __fastcall ServiceMain(__int64 a1, const wchar_t **serviceName)
{
    struct _SERVICE_STATUS ServiceStatus; // [rsp+20h] [rbp-258h] BYREF
    struct _SERVICE_STATUS v4; // [rsp+40h] [rbp-238h] BYREF
    WCHAR ServiceName[256]; // [rsp+60h] [rbp-218h] BYREF

    OutputDebugStringA("ServiceMain Load");
    wcsncpy(ServiceName, *serviceName, 0x100ui64);
    hServiceStatus = RegisterServiceCtrlHandlerW(ServiceName, (LPHANDLER_FUNCTION)HandlerProc);
    if ( hServiceStatus )
    {
        FreeConsole();
        ServiceStatus.dwServiceType = SERVICE_WIN32_SHARE_PROCESS;
        ServiceStatus.dwServiceSpecificExitCode = 0;
        currentServiceStatus = SERVICE_START_PENDING;
        ServiceStatus.dwCurrentState = SERVICE_START_PENDING;
        *(_QWORD *)&ServiceStatus.dwControlsAccepted = SERVICE_CONTINUE_PENDING;
        ServiceStatus.dwCheckPoint = 1;
        ServiceStatus.dwWaitHint = 1000;
        SetServiceStatus(hServiceStatus, &ServiceStatus);
        v4.dwServiceType = SERVICE_WIN32_SHARE_PROCESS;
        currentServiceStatus = SERVICE_RUNNING;
        v4.dwCurrentState = SERVICE_RUNNING;
        *(_QWORD *)&v4.dwControlsAccepted = SERVICE_CONTINUE_PENDING;
        *(_QWORD *)&v4.dwServiceSpecificExitCode = 0i64;
        v4.dwWaitHint = 1000;
        SetServiceStatus(hServiceStatus, &v4);
        Entry();
    }
    OutputDebugStringA("Exit service");
}
```

ServiceMain function pseudocode

`ServiceMain` takes a service name as an argument and attempts to register a service control handler with a specific `HandlerProc` function meant to check and set the status of that service. With a valid service status handle, Mail-O detaches the calling process from its console, changes the service status values to reflect its current running state, and calls the `Entry` function. Note the `ServiceMain` function with the debug string “ServiceMain Load” – a template that comes into play in looking for connections to other malware.

### Mail-O: Entry

The `Entry` function is called at the end of `ServiceMain`, but it can also be independently invoked. It checks for the presence of `%AllUsersProfile%PSEXESVC.EXE` and launches it as a process. This function is registered as a top level exception filter.

```
mov     [rsp][0000000E0],bx
call    .00000001`80009780 --↓2
lea     rdx,[rsp][0000000E0]
lea     rcx,[00000001`8001DAA0] ;'%AllUsersProfile%\PSEXESVC.EXE' --↓3
mov     r8d,000000104
call    ExpandEnvironmentStringsW
lea     rcx,[rsp][0000000E0]
call    PathFileExistsW
test    eax,eax
```

Mail-O PSEXESVC.exe check function

The main `Entry` logic is orchestrated in the next function. First, Mail-O checks the registry for an existing install of the legitimate Mail.Ru Disk-O software. It decrypts configuration strings and contacts

`https://dispatcher.cloud.mail.ru/` .

Mail-O uses the SystemTime to POST the encrypted victim hostname (or in its absence the string “[none]”) and receive a payload. The payload is written to a temporary path before being launched. Mail-O then goes into a sleep loop until a predetermined amount of time.

We’ve yet to see ‘Webdav-O’, the other malware component described in the Rostelecom-Solar report. However, that shouldn’t keep us from following an interesting lead.

## The ‘Entry’ Connection

```
int 3
push 01001873C ;'DLL Entry:' -->1
call OutputDebugStringA
push 010001A20 -->2
call SetUnhandledExceptionFilter
push 000008002 ;' Çe'
call SetErrorMode
call .010001850 -->3
xor eax,eax
ret ; --A-A-A-A-A-A-A-A-A-A-A-A-A-A-A-A-
int 3
push ebp
mov ebp,esp
sub esp,000000238 ;' 08'
push edi
mov edi,OutputDebugStringA -->4
push 010018748 ;'ServiceMain Load' -->5
call edi
mov eax,[ebp][00C]
push 00000100
push d,[eax]
lea eax,[ebp][-00000238]
push 00000100
push eax
call .01000828E -->6
add esp,010
lea eax,[ebp][-00000238]
push 010001BD0 -->7
push eax
call RegisterServiceCtrlHandlerW
mov [010031170],eax
test eax,eax
jnz .010001ABC -->8
push 01001875C ;'SvcHostDLL RegisterServiceCtrlH.
call edi
pop edi
mov esp,ebp
pop ebp
ret ; --A-A-A-A-A-A-A-A-A-A-A-A-A-A-A-A-
8call FreeConsole
lea eax,[ebp][-01C]
mov d,[ebp][-01C],00000010
push eax
sub rsp,038 ;'8'
lea rcx,[00000001`8007F500] -->1
call SetUnhandledExceptionFilter
mov ecx,000008002 ;' Çe'
call SetErrorMode
nop
call .00000001`8007F980 -->2
nop
call .00000001`8007F4A0 -->3
xor ecx,ecx
call ExitProcess
nop
add rsp,038 ;'8'
ret ; --A-A-A-A-A-A-A-A-A-A-A-A-A-A-A-A-
int 3
push rbx
sub rsp,00000270 ;' 0p'
mov rax,[00000001`802B5690] -->4
xor rax,rsi
mov [rsp][00000260],rax
lea rcx,[00000001`802989E8] ;'ServiceMain Load' --
mov rdx,rcx
call OutputDebugStringA
mov rdx,[rbx]
lea rcx,[rsp][060]
mov r8d,00000100
call .00000001`801FAAD8 -->6
lea rdx,[00000001`8007F770] -->7
lea rcx,[rsp][060]
call RegisterServiceCtrlHandlerW
[00000001`802BABB0],rax
test rax,rax
jnz .00000001`8007F662 -->8
lea rcx,[00000001`80298A00] ;'Exit service' -->9
call OutputDebugStringA
mov rcx,[rsp][00000260]
xor rcx,rsi
call .00000001`801EC530 -->A
add rsp,00000270 ;' 0p'
pop rbx
ret ; --A-A-A-A-A-A-A-A-A-A-A-A-A-A-A-A-
8call FreeConsole
mov rcx,[00000001`802BABB0]
lea rdx,[rsp][020]
xor ebx,ebx
```

Left: TManger sample (NTT Security)

71fe3edbee0c27121386f9c01b723e1cfb416b7af093296bd967bbabdc706393

Right: Mail-O sample:

603881f4c80e9910ab22f39717e8b296910bff08cd0f25f78d5bff1ae0dce5d7

Mail-O exports a function called `Entry`, presumably a misspelling of ‘Entry’. Misspellings are a true gift for malware researchers. As it turns out, this isn’t the first time that misspelling has been noted in a recently deployed piece of malware.

In December 2020, Ignacio Sanmillan and Matthieu Faou released an excellent report on a Vietnamese supply-chain attack that used PhantomNet (aka SManager) malware. The researchers noted that the malware’s persistence was established via a scheduled task that called the malicious DLL’s export, ‘Entry’. The researchers note that this same export was pointed out by [NTT Security](#) in their analysis of TManger malware, which they in turn

correlate with Proofpoint's 'TA428' threat actor. That nondescript threat actor name is adopted by [Dr. Web in reporting](#) recent attacks against additional Russian targets including research institutes.

While that might all seem a bit convoluted, I rehearse the logical connections to illustrate two points:

1. There's an established history of this very non-Western 'threat actor' in targeting both Asian and Russian targets.
2. These presumably Chinese clusters of activity are confusing and difficult to disentangle. Tooling is likely shared among multiple threat actors (likely including PhantomNet/SManager), and what's being referred to as 'TA428' is probably an amalgam of multiple threat groups.

For skeptics, we've provided a YARA rule below for the `Entery` overlap, which entails not just the export function name but also the general layout of the function and some shared strings. Note that the layout has likely developed iteratively from an open-source template.

Finally, while I'm quick to disparage the quality of the malware as not up to some exalted Western standard, it's important to note that ThunderCats (and the larger TA428 umbrella) are pulling off custom-tailored region-specific supply chain attacks, successfully punching way above their weight in their intelligence collection efforts, and they should not be underestimated as an adversary.

## YARA

```
import "pe"

rule apt_CN_ThunderCats_Overlap
{
    meta:
        desc = "Thundercats Entery Export Overlap"
        author = "JAG-S @ SentinelLabs"
        version = "1.0"
        last_modified = "06.08.2021"
        reference = "https://rt-solar.ru/upload/iblock/b55/Ataki-na-FOIV_otchet-NKTSKI-i-Rostelekom_!"

    strings:
        $psexesvc = "%AllUsersProfile%PSEXESVC.EXE" ascii wide
        $sm_load = "ServiceMain Load" ascii wide fullword

    condition:
        uint16(0) == 0x5a4d
        and
        pe.exports("Entery")
        and
        pe.exports("ServiceMain")
        and
        all of them
}
```

## References

<https://www.bbc.com/news/world-europe-36933239>

<https://www.reuters.com/technology/russias-fsb-reports-unprecedented-hacking-campaign-aimed-government-agencies-2021-05-26/>

<https://rt-solar.ru/analytics/reports/2203/>

<https://st.drweb.com/static/new->

[www/news/2021/april/drweb\\_research\\_attacks\\_on\\_russian\\_research\\_institutes\\_en.pdf](www/news/2021/april/drweb_research_attacks_on_russian_research_institutes_en.pdf)

<https://www.welivesecurity.com/2020/12/17/operation-signsight-supply-chain-attack-southeast-asia/>

---

Source: <https://labs.sentinelone.com/thundercats-hack-the-fsb-your-taxes-didnt-pay-for-this-op/>