Chinese ‘Space Pirates’ are hacking Russian aerospace firms

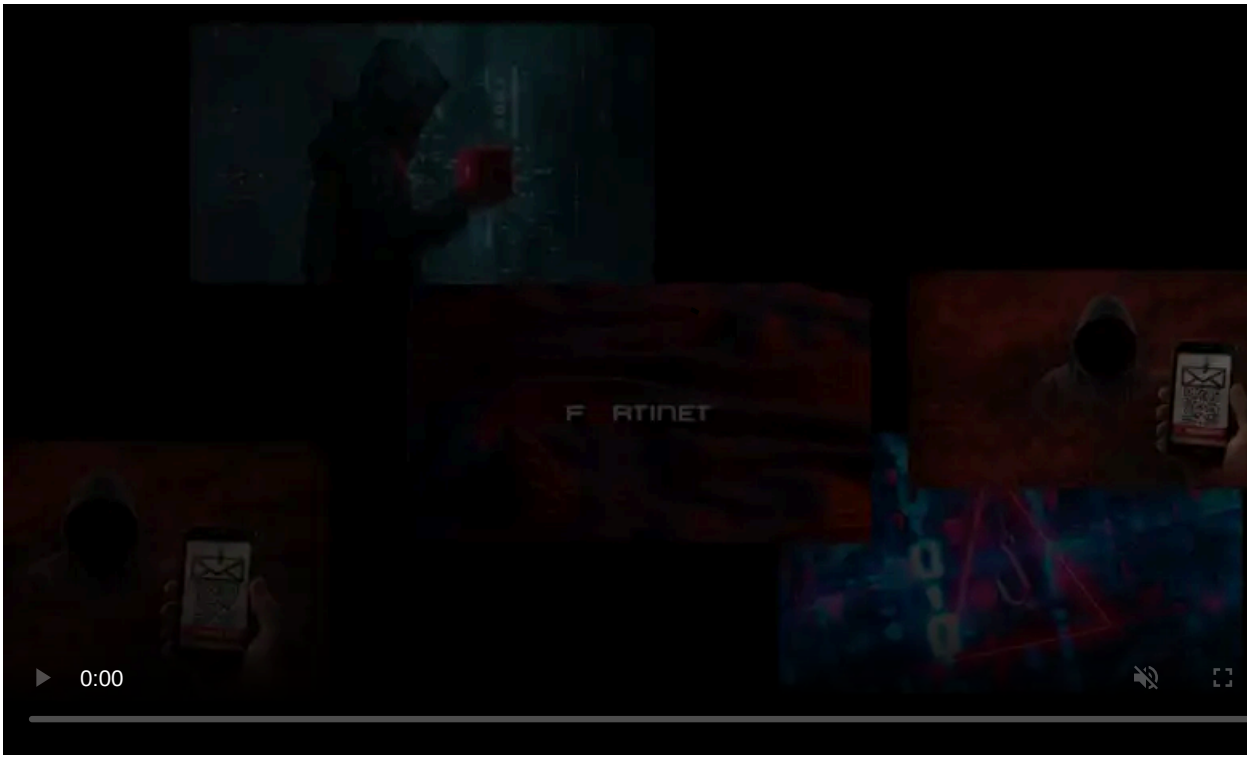
By Bill Toulas

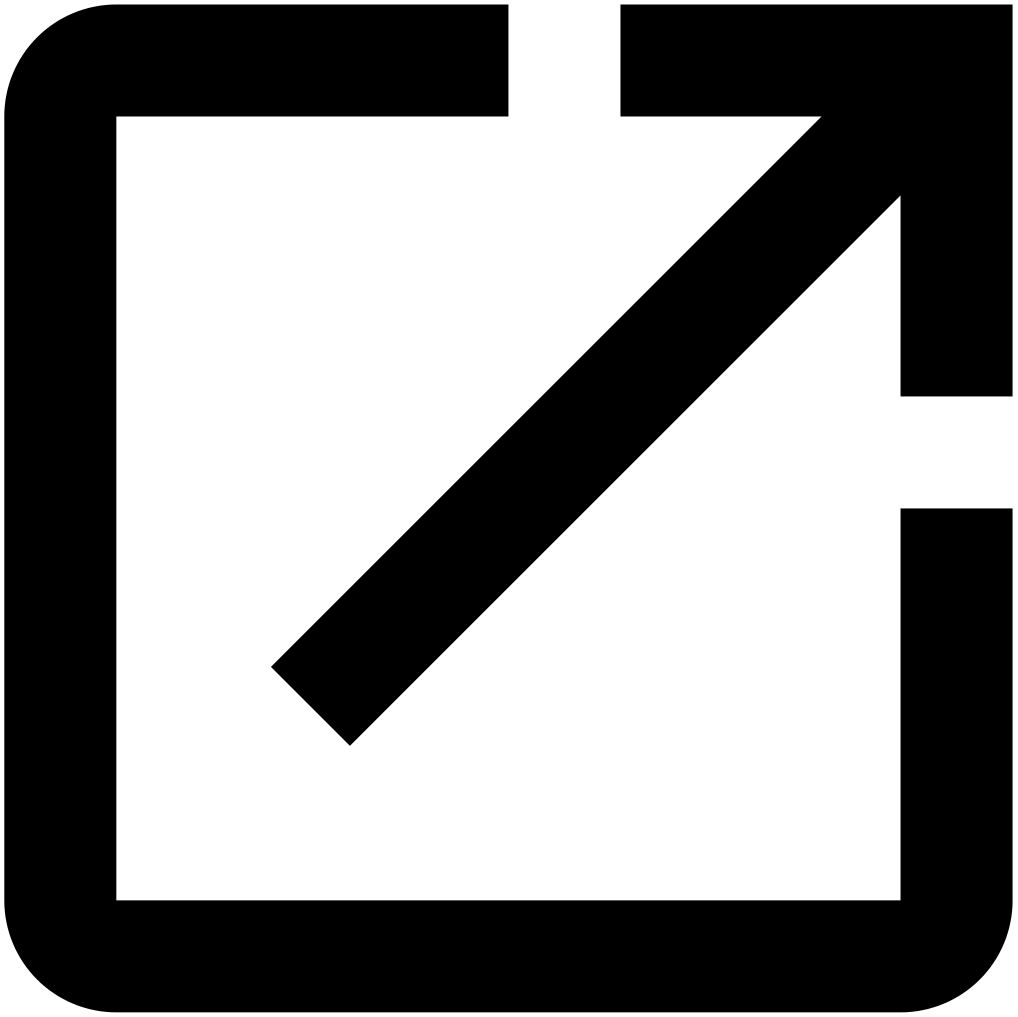
Published: 2022-05-18 · Archived: 2026-04-06 03:31:47 UTC

A previously unknown Chinese hacking group known as 'Space Pirates' targets enterprises in the Russian aerospace industry with phishing emails to install novel malware on their systems.

The threat group is believed to have started operating in 2017, and while it has links to known groups like APT41 (Winnti), Mustang Panda, and APT27, it is thought to be a new cluster of malicious activity.

Russian threat analysts at Positive Technologies named the group "Space Pirates" due to their espionage operations focusing on stealing confidential information from companies in the aerospace field.


F RTINET
0:00



In the wild detections

The Space Pirates APT group has been seen targeting government agencies and enterprises involved in IT services, aerospace, and electric power industries located in Russia, Georgia, and Mongolia.

The threat analysts first discovered signs of Space Pirates' activity last summer during incident response and quickly confirmed that the threat actors used the same malware and infrastructure against at least four more domestic entities since 2019.

Two of these cases concern Russian companies with state participation, which the hackers successfully compromised.

In the first case, the threat actors maintained their access to 20 servers for ten months, stealing over 1,500 documents, employee details, and other sensitive data.

In the second case, the Chinese hackers stayed in the network of the compromised company for over a year, siphoning confidential information and installing their malware to 12 corporate network nodes in three distinct regions.

Novel malware

The arsenal of Space Pirates consists of custom loaders hiding behind decoy documents, slightly modified backdoors that have been around for years, the Chinese trademark malware PlugX, and tailored spins of the PcShare backdoor.

Moreover, Space Pirates' attacks have also employed ShadowPad, Zupdax, PoisonIvy, and ReVBShell in attacks.

In addition to the above, the newly discovered APT uses three previously undocumented modular malware tools, namely Deed RAT, BH_A006, and MyKLoadClient.

MyKLoadClient is a loader using SFX archives combined with DLL side-loading through an auxiliary launcher library signed by McAfee Inc. The launcher supports commands that give the threat actors close control over the infection.

BH_A006 is a heavily modified version of the Gh0st backdoor, featuring many layers of obfuscation to bypass security protections and thwart analysis.

Its features include network service creation, UAC bypassing, and shellcode unpacking and launching in the memory.

```
18  if ( !load_imports() )
19    return -1;
20  (imports.kernel32_SetErrorMode)(2);
21  memset(v14, 0, sizeof(v14));
22  (imports.kernel32_GetModuleFileNameW)(0, v14, 260);
23  v1 = (imports.kernel32_GetCommandLineW)();
24  debug(L"Argv12%s", v1);
25  debug(L"GetModuleFileName %s", v14);
26  if ( wcsstr(v1, L"InsertS") )
27  {
28    create_and_start_service();
29    (imports.kernel32_Sleep)(1000);
30    v2 = (imports.kernel32_GetCurrentProcess)(0);
31    (imports.kernel32_TerminateProcess)(v2);
32  }
33  else if ( wcsstr(v1, L"runsvc") )
34  {
35    debug(L"svchost.exe %s", v14);
36    kernel32_DeleteFileA = imports.kernel32_DeleteFileA;
37    (imports.kernel32_DeleteFileA)("C:\\ProgramData\\Sandboxie\\SbieMsg.dll");
38    kernel32_DeleteFileA("C:\\ProgramData\\Sandboxie\\SbieMsg.dat");
39    kernel32_DeleteFileA("C:\\Windows \\System32\\SSPICLI.dll");
40    kernel32_DeleteFileA("C:\\Windows \\System32\\perfmon.exe");
41    kernel32_DeleteFileA("C:\\Windows \\System32\\dxva2.dll");
42    kernel32_DeleteFileA("C:\\Windows \\System32\\dccw.exe");
43    kernel32_RemoveDirectoryA = imports.kernel32_RemoveDirectoryA;
44    (imports.kernel32_RemoveDirectoryA)("C:\\Windows \\System32\\");
45    kernel32_RemoveDirectoryA("C:\\Windows \\");
46    kernel32_CreateThread = imports.kernel32_CreateThread;
47    (imports.kernel32_CreateThread)(0, 0, inject_payload, 0, 0, 0);
48    (imports.kernel32_Sleep)(2000);
49    kernel32_CreateThread(0, 0, check_mapping, 0, 0, 0);
50  }
51  else if ( wcsstr(v1, L"ByPassUAC") )
52  {
53    v6 = alloc_and_lock(dword_1387B78);
54    memset(v6, 0, dword_1387B78);
```

BH_A006 shellcode loading *(PT)*

Another interesting custom tool is Deed RAT, which features an unusual, intelligent method of transferring control to the shellcode.

Deed RAT's functions depend on which plugins are fetched and loaded. For example, PT has seen eight plugins for startup, C2 config, installation, code injection into processes, network interactions, connection management, registry editing, registry monitoring, and proxy sniffing.

The supported protocols for C2 communication include TCP, TLS, HTTP, HTTPS, UDP, and DNS, so there's generally a high level of versatility.

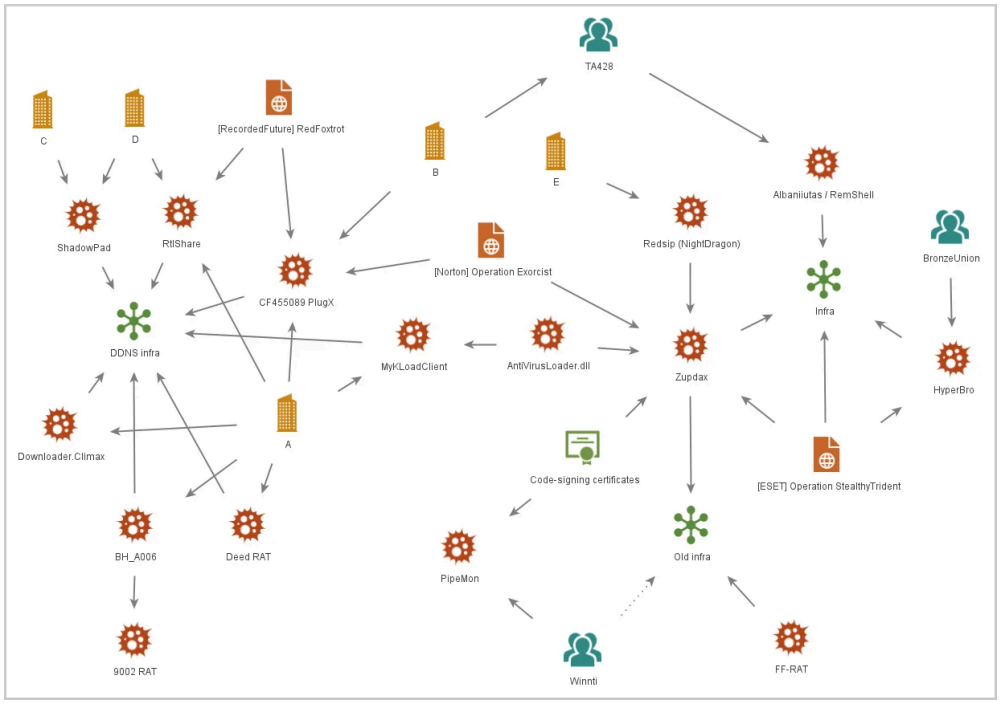
The commands supported by Deed RAT are the following:

- Collect system information
- Create a separate communication channel for a plugin
- Self-remove
- Ping
- Deactivate connection
- Update the shellcode for an injection stored in the registry
- Update the main shellcode on disk and delete all plugins

Chinese convolution

The threat analysts believe that the overlaps between various Chinese APTs are due to tool exchanges, a common phenomenon for hackers in the region.

Using shared tools further obscures the traces of distinct threat groups and makes the work of analysts a lot harder, so Chinese APTs have multiple reasons to follow this practice.



Various links between Chinese APTs *(PT)*

Space Pirates has also been seen deploying their custom malware on some Chinese firms for financial gains, so the threat group might have a dual function.

Chinese hackers have been very aggressive against Russian targets lately, as confirmed by recent findings of analysts at Secureworks and Google.

Espionage is a standard operation for Chinese APTs, and Russia is a valid target that excels in aerospace, weapons, electrical engineering, shipbuilding, and nuclear technology.

Automated Pentesting Covers Only 1 of 6 Surfaces.

Automated pentesting proves the path exists. BAS proves whether your controls stop it. Most teams run one without the other.

This whitepaper maps six validation surfaces, shows where coverage ends, and provides practitioners with three diagnostic questions for any tool evaluation.
