

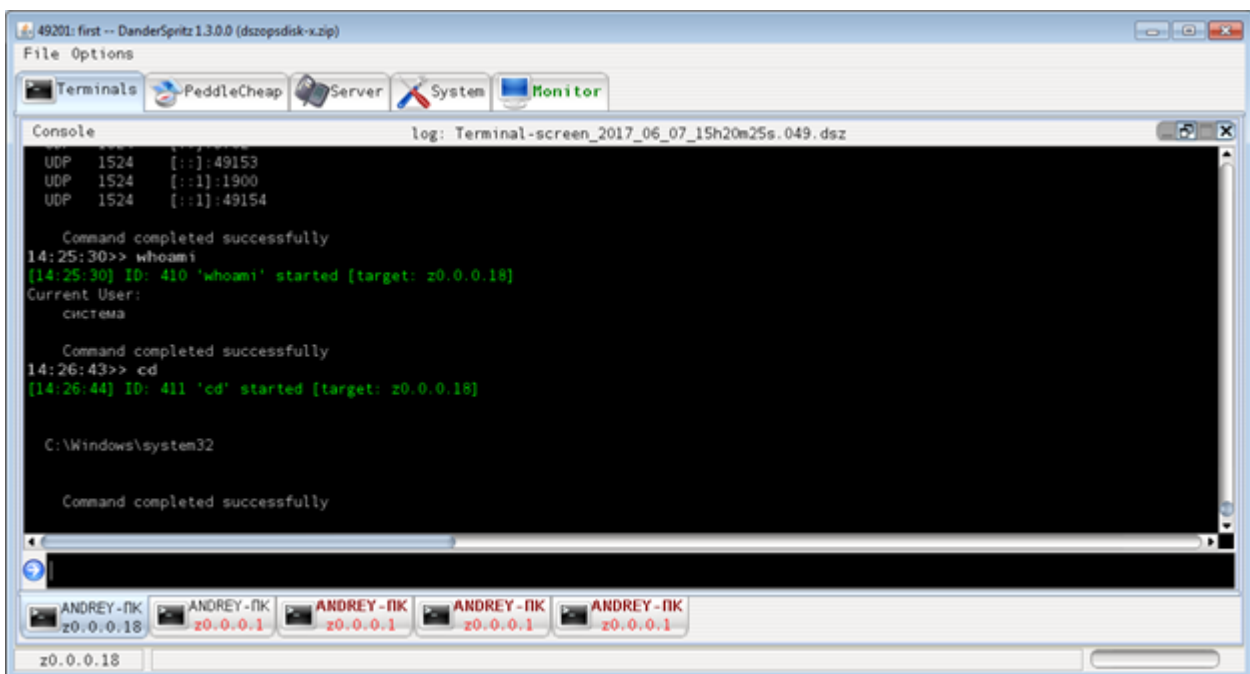
# DarkPulsar

By Andrey Dolgushev

Published: 2018-10-19 · Archived: 2026-04-02 12:19:09 UTC

In March 2017, the ShadowBrokers published a chunk of stolen data that included two frameworks: DanderSpritz and FuzzBunch.

DanderSpritz consists entirely of plugins to gather intelligence, use exploits and examine already controlled machines. It is written in Java and provides a graphical windows interface similar to botnets administrative panels as well as a Metasploit-like console interface. It also includes its own backdoors and plugins for not-FuzzBunch-controlled victims.



## DanderSpritz interface

Fuzzbunch on the other hand provides a framework for different utilities to interact and work together. It contains various types of plugins designed to analyze victims, exploit vulnerabilities, schedule tasks, etc. There are three files in the plugin set from the FuzzBunch framework:

- %pluginName%-version.fb

This is the utility file of the framework. It duplicates the header from XML and includes the plugin's ID.

- %pluginName%-version.exe

This executable file is launched when FuZZbuNch receives the command to do so.

- %pluginName%-version.xml

This configuration file describes the plugin's input and output parameters – the parameter name, its type and description of what it's responsible for; all of these can be shown in FuzzBunch as a prompt. This file also contributes a lot to the framework's usability, as it supports the specification of default parameters.

One of the most interesting Fuzzbunch's categories is called ImplantConfig and includes plugins designed to control the infected machines via an implant at the post-exploitation stage. **DarkPulsar** is a very interesting administrative module for controlling a passive backdoor named 'sipauth32.tsp' that provides remote control, belonging to this category.

It supports the following commands:

- Burn
- RawShellcode
- EDFStagedUpload
- DisableSecurity
- EnableSecurity
- UpgradeImplant
- PingPong

*Burn*, *RawShellcode*, *UpgradeImplant*, and *PingPong* remove the implant, run arbitrary code, upgrade the implant and check if the backdoor is installed on a remote machine, respectively. The purpose of the other commands is not that obvious and, to make it worse, the leaked framework contained only the administrative module to work with DarkPulsar's backdoor, but not the backdoor itself.

While analyzing the administrative module, we noticed several constants that are used to encrypt the traffic between the C&C and the implant:

```
(TcLog)(v2, 5, "[+] - Performing crypto session setup\n");
v3 = v1[1];
sub_402B70(pbBuffer, 4u);
*pbBuffer[4] = 0x3BA6814F - *pbBuffer;
v4 = *pbBuffer ^ (0x3BA6814F - *pbBuffer);
v5 = *v1;
*(&v28 + 1) = 4;
HIBYTE(v27) = 5;
*(&v27 + 3) ^= v4;
*(&v28 + 3) = v4 ^ 0xAA64F13D;
v21 = 16;
v20 = 16;
v22 = pbBuffer;
v6 = (*(*v5 + 8))(&v20, &v23);
v7 = v6;
if ( v6 && v6 != 0x90312 )
{
    TcLog(v1[2], 3, "[%s] - CDPPProtocolHandler::SendRecv Failed (0x%x)\n",
        "CDPClient::PerformSetupSession", v6);
}
else
{
    v8 = v25;
    if ( (v25 || v23) && v23 >= 16 )
    {
        v9 = *v25;
        v16 = v25;
        if ( *v25 + v25[1] == 0xA13C82E )
```

We thought that probably these constants should also appear in the backdoor, so we created a detection for them. Several months later we found our mysterious DarkPulsar backdoor. We later were able to find both 32- and 64-bit versions.

We found around 50 victims located in Russia, Iran and Egypt, typically infecting Windows 2003/2008 Server. Targets were related to nuclear energy, telecommunications, IT, aerospace and R&D.

## DarkPulsar technical highlights

The DarkPulsar implant is a dynamic library whose payload is implemented in exported functions. These functions can be grouped as follows:

1. 1 Two nameless functions used to install the backdoor in the system.
2. 2 Functions with names related to TSPI (Telephony Service Provider Interface) operations that ensure the backdoor is in the autorun list and launched automatically.
3. 3 A function with a name related to SSPI (Security Support Provider Interface) operations. It implements the main malicious payload.

The implementations of the SSPI and TSPI interfaces are minimalistic: the functions that are exported by DarkPulsar have the same names as the interface functions; however, they include malicious code instead of the phone service.

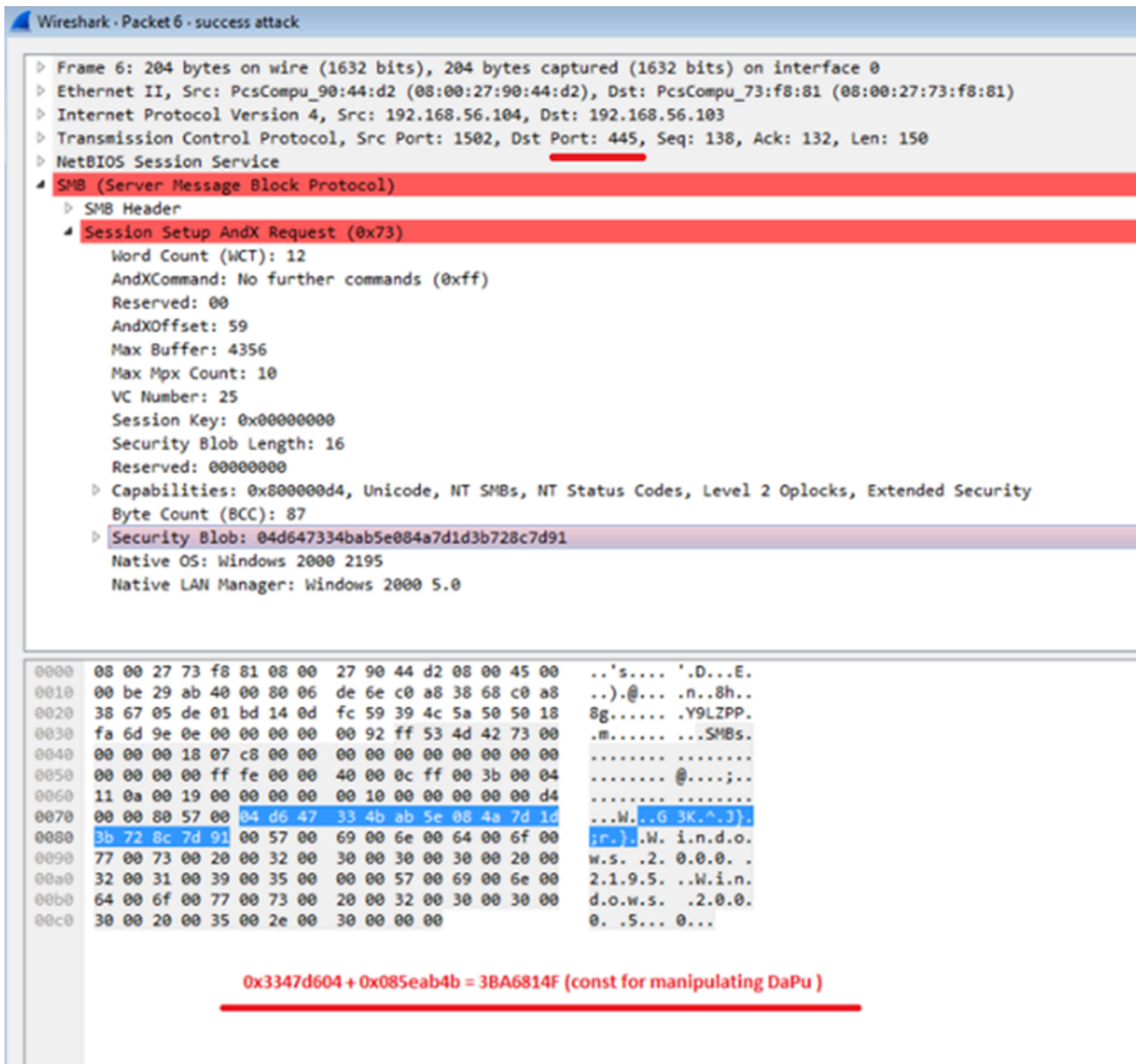
The implant is installed in the system by the nameless exported function. The backdoor is launched by calling Secur32.AddSecurityPackage with administrator privileges with the path to its own library in the parameter, causing lsass.exe to load DarkPulsar as SSP/AP and to call its exported function *SpLsaModeInitialize* used by DarkPulsar to initialize the backdoor. In this way AddSecurityPackage is used to inject code into lsass.exe. It also adds its library name at HKLM\Software\Microsoft\Windows\CurrentVersion\Telephony\Providers

This is loaded at start by the Telephony API (TapiSrv) launched alongside the Remote Access Connection Manager (RasMan) service, setting startup type as “Automatic”. When loading the telephony service provider’s library, TapiSrv calls *TSPI\_lineNegotiateTSPIVersion* which contains the AddSecurityPackage call to make the inject into lsass.exe.

DarkPulsar implements its payload by installing hooks for the SpAcceptLsaModeContext – function responsible for authentication. Such injects are made in several system authentication packets within the process lsass.exe and allow DarkPulsar to control authentication process based on the following protocols:

- Msv1\_0.dll – for the NTLM protocol,
- Kerberos.dll – for the Kerberos protocol,
- Schannel.dll – for the TLS/SSL protocols,
- Wdigest.dll – for the Digest protocol, and
- Lsasrv.dll –for the Negotiate protocol.

After this, Darkpulsar gets ability to embed malware traffic into system protocols. Since this network activity takes place according to standard system charts, it will only be reflected in the System process – it uses the system ports reserved for the above protocols without hindering their normal operation.



*Network traffic during successful connection to DarkPulsar implant*

The second advantage of controlling authentication processes is ability to bypass entering a valid username and password for obtaining access to objects that require authentication such as processes list, remote registry, file system through SMB. After Darkpulsar’s DisableSecurity command is sent, backdoor’s hooks on the victim side will always returns in the SpAcceptLsaModeContext function that passed credentials are valid. Getting that, system will provide access to protected objects to client.

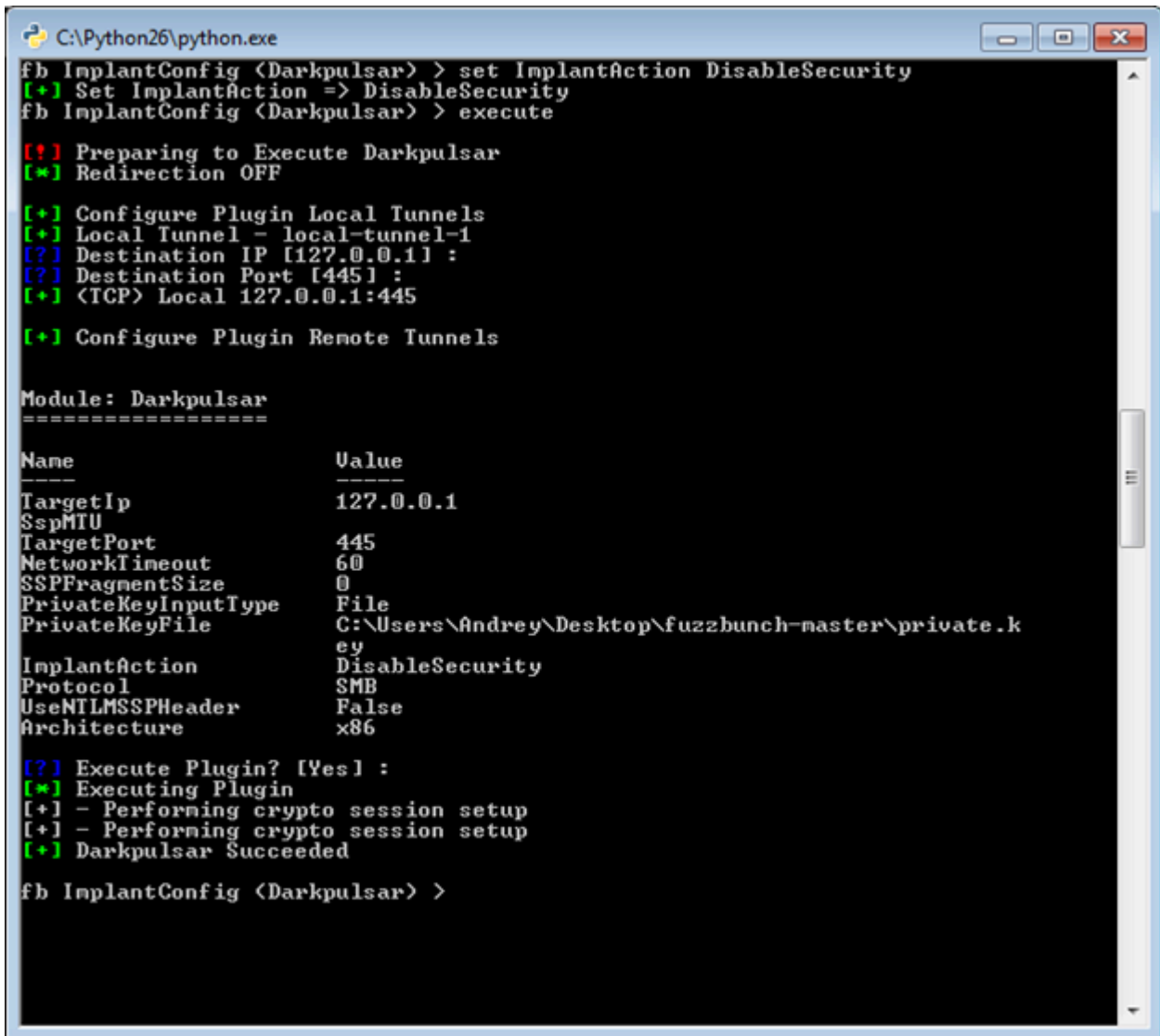
### Working with DarkPulsar

Darkpulsar-1.1.0.exe is the administrative interface working under the principle of “one command – one launch”. The command to be executed must be specified either in the configuration file Darkpulsar-1.1.0.9.xml or as command line arguments, detailing at least:

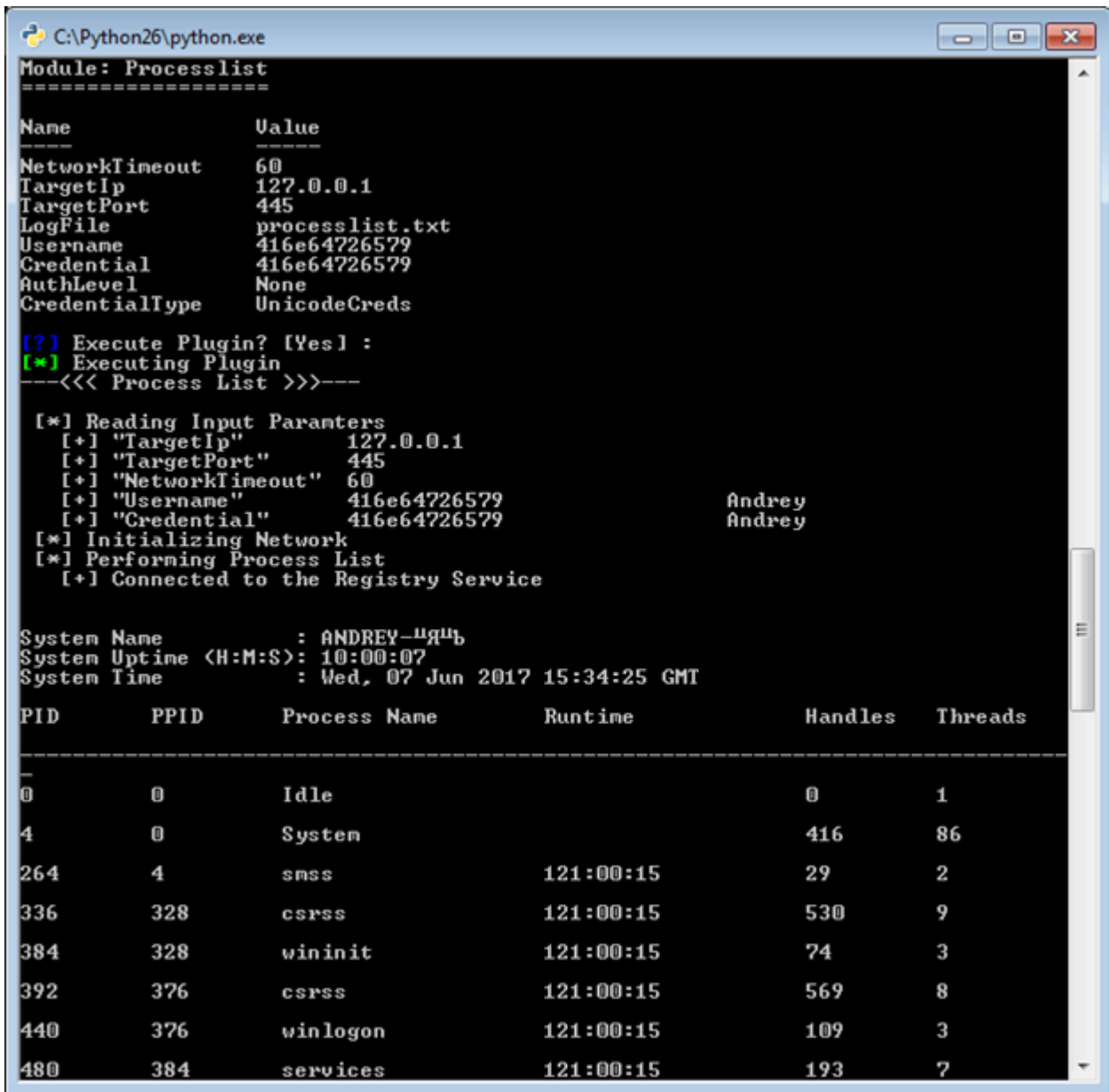
- whether the target machine uses a 32-bit or 64-bit system;

- protocol (SMB, NBT, SSL, RDP protocols are supported) to deliver the command and port number
- private RSA key to decrypt the session AES key

Darkpulsar-1.1.0 was not designed as a standalone program for managing infected machines. This utility is a plugin of the Fuzzbunch framework that can manage parameters and coordinate different components. Here is how DisableSecurity command in Fuzzbunch looks like:



Below is an example of Processlist after DisableSecurity, allowing to execute any plugin without valid credentials and operating via regular system functions (remote registry service):



## DanderSpritz

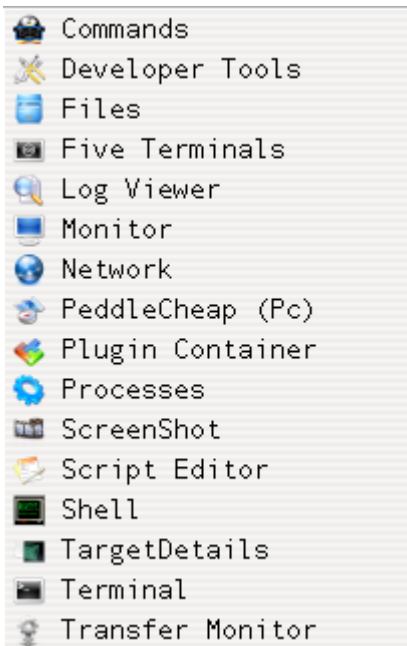
DanderSpritz is the framework for controlling infected machines, different from FuZZbuNch as the latter provides a limited toolkit for the post-exploitation stage with specific functions such as DisableSecurity and EnableSecurity for DarkPulsar.

For DanderSpritz works for a larger range of backdoors, using PeddleCheap in the victim to enable operators launching plugins. PeddleCheap is a plugin of DanderSpritz which can be used to configure implants and connect to infected machines. Once a connection is established all DanderSpritz post-exploitation features become available.

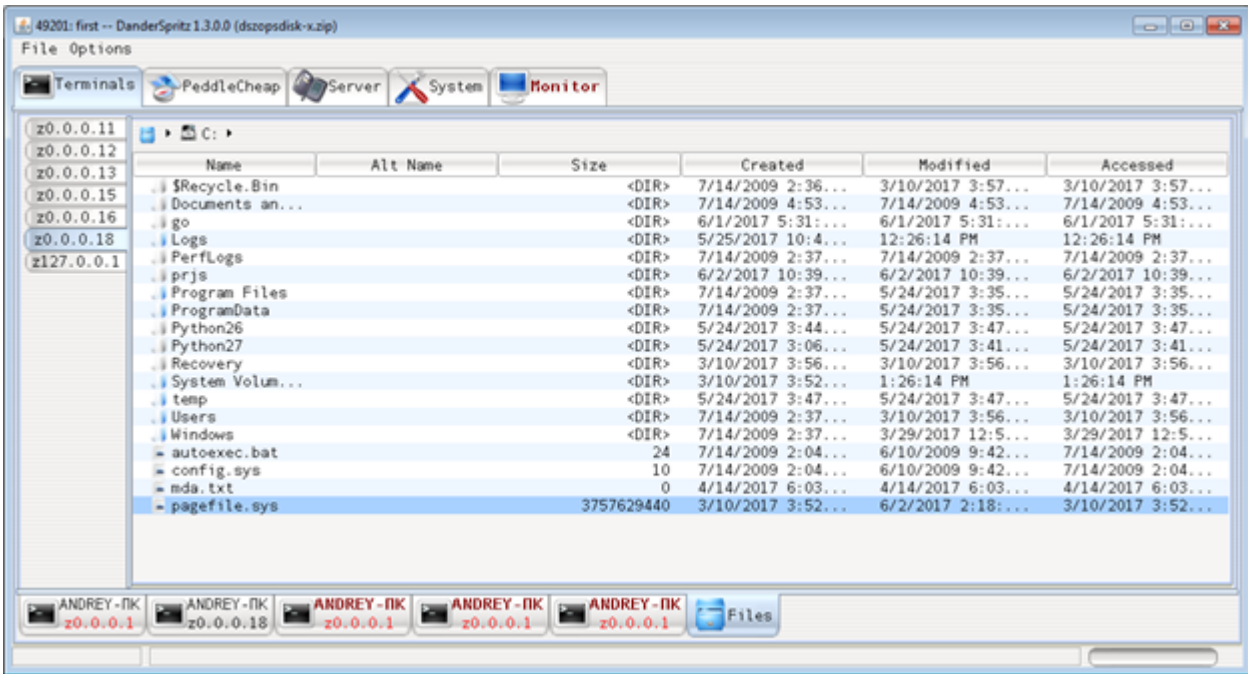
This is how DarkPulsar in EDFStagedUpload mode provides the opportunity to infect the victim with a more functional implant: PCDllLauncher (Fuzzbunch's plugin) deploys the PeddleCheap implant on the victim side, and DanderSpritz provides a user-friendly post-exploitation interface. Hence, the full name of PCDllLauncher is 'PeddleCheap DLL Launcher'.

The complete DanderSpritz usage scheme with the plugin PeddleCheap via FuZZbuNch with the plugins DarkPulsar and PCDllLauncher consists of four steps:

- Via FuZZbuNch, run command EDFStagedUpload to launch DarkPulsar.
- In DanderSpritz, run command pc\_prep (PeedelCheap Preparation) to prepare the payload and the library to be launched on the implant side.
- In DanderSpritz, run command pc\_old (which is the alias of command pc\_listen -reuse -nolisten -key Default) – this sets it to wait for a socket from Pcdlllauncher.
- Launch Pcdlllauncher via FuZZbuNch and specify the path to the payload that has been prepared with the command pc\_prep in the ImplantFilename parameter.



*DanderSpritz*



File System plugin

## Conclusions

The FuzzBunch and DanderSpritz frameworks are designed to be flexible and to extend functionality and compatibility with other tools. Each of them consists of a set of plugins designed for different tasks: while FuzzBunch plugins are responsible for reconnaissance and attacking a victim, plugins in the DanderSpritz framework are developed for managing already infected victims.

The discovery of the DarkPulsar backdoor helped in understanding its role as a bridge between the two leaked frameworks, and how they are part of the same attacking platform designed for long-term compromise, based on DarkPulsar's advanced abilities for persistence and stealthiness. The implementation of these capabilities, such as encapsulating its traffic into legitimate protocols and bypassing entering credentials to pass authentication, are highly professional.

Our product can completely remove the related to this attack malware.

## Detecting malicious network activity

When EDFStagedUpload is executed in an infected machine, a permanent connection is established, which is why traffic via port 445 appears. A pair of bound sockets also appears in lsass.exe:

| Proc...     | PID  | Protocol | Local Address | Local Port   | Remote Address | Remote Port | State       | Sent Packets | Sent Bytes | Rcvd Packets | Rcvd Bytes |
|-------------|------|----------|---------------|--------------|----------------|-------------|-------------|--------------|------------|--------------|------------|
| alg.exe     | 1632 | TCP      | work          | 1027         | work           | 0           | LISTENING   |              |            |              |            |
| igmp.exe    | 632  | TCP      | work          | 5152         | work           | 0           | LISTENING   |              |            |              |            |
| lsass.exe   | 708  | UDP      | work          | 4500         | *              | *           |             |              |            |              |            |
| lsass.exe   | 708  | TCP      | work          | 1037         | localhost      | 1038        | ESTABLISHED |              |            |              |            |
| lsass.exe   | 708  | TCP      | work          | 1038         | localhost      | 1037        | ESTABLISHED |              |            |              |            |
| svchost.exe | 996  | TCP      | work          | epnmap       | work           | 0           | LISTENING   |              |            |              |            |
| svchost.exe | 1296 | UDP      | work          | 1900         | *              | *           |             |              |            | 20           | 3,152      |
| svchost.exe | 1116 | UDP      | work          | 1025         | *              | *           |             |              |            |              |            |
| svchost.exe | 1116 | UDP      | work          | ntp          | *              | *           |             |              |            |              |            |
| svchost.exe | 1116 | UDP      | work          | ntp          | *              | *           |             |              |            |              |            |
| svchost.exe | 1296 | UDP      | work          | 1900         | *              | *           |             |              |            |              |            |
| System      | 4    | TCP      | work          | netbios-ssn  | work           | 0           | LISTENING   |              |            |              |            |
| System      | 4    | TCP      | work          | microsoft-ds | work           | 0           | LISTENING   |              |            |              |            |
| System      | 4    | UDP      | work          | netbios-ns   | *              | *           |             | 32           | 1,887      | 114          | 4,350      |
| System      | 4    | UDP      | work          | netbios-dgm  | *              | *           |             | 19           | 3,618      | 30           | 179        |
| System      | 4    | UDP      | work          | microsoft-ds | *              | *           |             |              |            |              |            |
| System      | 4    | TCP      | work          | microsoft-ds | andreyAb       | 49176       | ESTABLISHED | 40           | 6,576      | 40           | 9,312      |

Endpoints: 18    Established: 3    Listening: 5    Time Wait: 0    Close Wait: 0

When DanderSpritz deploys PeddleCheap’s payload via the PcDllLauncher plugin, network activity increases dramatically:

| Proc...     | PID  | Protocol | Local Address | Local Port   | Remote Address | Remote Port | State       | Sent Packets | Sent Bytes | Rcvd Packets | Rcvd Bytes |
|-------------|------|----------|---------------|--------------|----------------|-------------|-------------|--------------|------------|--------------|------------|
| alg.exe     | 1632 | TCP      | work          | 1027         | work           | 0           | LISTENING   |              |            |              |            |
| igmp.exe    | 632  | TCP      | work          | 5152         | work           | 0           | LISTENING   |              |            |              |            |
| lsass.exe   | 708  | UDP      | work          | 4500         | *              | *           |             |              |            |              |            |
| lsass.exe   | 708  | TCP      | work          | 1037         | localhost      | 1038        | ESTABLISHED | 313          | 62,494     | 410          | 830,153    |
| lsass.exe   | 708  | TCP      | work          | 1038         | localhost      | 1037        | ESTABLISHED | 537          | 840,761    | 312          | 62,494     |
| svchost.exe | 996  | TCP      | work          | epnmap       | work           | 0           | LISTENING   |              |            |              |            |
| svchost.exe | 1296 | UDP      | work          | 1900         | *              | *           |             |              |            | 32           | 5,240      |
| svchost.exe | 1116 | UDP      | work          | 1025         | *              | *           |             |              |            |              |            |
| svchost.exe | 1116 | UDP      | work          | ntp          | *              | *           |             |              |            |              |            |
| svchost.exe | 1116 | UDP      | work          | ntp          | *              | *           |             |              |            |              |            |
| svchost.exe | 1296 | UDP      | work          | 1900         | *              | *           |             |              |            |              |            |
| System      | 4    | TCP      | work          | netbios-ssn  | work           | 0           | LISTENING   |              |            |              |            |
| System      | 4    | TCP      | work          | microsoft-ds | work           | 0           | LISTENING   |              |            |              |            |
| System      | 4    | UDP      | work          | netbios-ns   | *              | *           |             | 68           | 3,687      | 240          | 10,650     |
| System      | 4    | UDP      | work          | netbios-dgm  | *              | *           |             | 22           | 4,229      | 35           | 179        |
| System      | 4    | UDP      | work          | microsoft-ds | *              | *           |             |              |            |              |            |
| System      | 4    | TCP      | work          | microsoft-ds | andreyAb       | 49176       | ESTABLISHED | 1,704        | 299,816    | 1,900        | 1,218,392  |

Endpoints: 18    Established: 3    Listening: 5    Time Wait: 0    Close Wait: 0

When a connection to the infected machine is terminated, network activity ceases, and only traces of the two bound sockets in lsass.exe remain:

| Proc...     | PID  | Protocol | Local Address | Local Port   | Remote Address | Remote Port | State       | Sent Packets | Sent Bytes | Rcvd Packets | Rcvd Bytes |
|-------------|------|----------|---------------|--------------|----------------|-------------|-------------|--------------|------------|--------------|------------|
| alg.exe     | 1632 | TCP      | work          | 1027         | work           | 0           | LISTENING   |              |            |              |            |
| igmp.exe    | 632  | TCP      | work          | 5152         | work           | 0           | LISTENING   |              |            |              |            |
| lsass.exe   | 708  | UDP      | work          | 4500         | *              | *           |             |              |            |              |            |
| lsass.exe   | 708  | TCP      | work          | 1037         | localhost      | 1038        | ESTABLISHED | 1,151        | 246,538    | 1,070        | 1,149,811  |
| lsass.exe   | 708  | TCP      | work          | 1038         | localhost      | 1037        | ESTABLISHED | 1,215        | 1,166,277  | 1,114        | 246,538    |
| svchost.exe | 996  | TCP      | work          | epnmap       | work           | 0           | LISTENING   |              |            |              |            |
| svchost.exe | 1296 | UDP      | work          | 1900         | *              | *           |             |              |            | 40           | 6,632      |
| svchost.exe | 1116 | UDP      | work          | 1025         | *              | *           |             |              |            |              |            |
| svchost.exe | 1116 | UDP      | work          | ntp          | *              | *           |             |              |            |              |            |
| svchost.exe | 1116 | UDP      | work          | ntp          | *              | *           |             |              |            |              |            |
| svchost.exe | 1296 | UDP      | work          | 1900         | *              | *           |             |              |            |              |            |
| System      | 4    | TCP      | work          | netbios-ssn  | work           | 0           | LISTENING   |              |            |              |            |
| System      | 4    | TCP      | work          | microsoft-ds | work           | 0           | LISTENING   |              |            |              |            |
| System      | 4    | UDP      | work          | netbios-ns   | *              | *           |             | 71           | 4,020      | 420          | 19,500     |
| System      | 4    | UDP      | work          | netbios-dgm  | *              | *           |             | 24           | 4,609      | 38           | 179        |
| System      | 4    | UDP      | work          | microsoft-ds | *              | *           |             |              |            |              |            |

Endpoints: 17    Established: 2    Listening: 5    Time Wait: 0    Close Wait: 0

## IOCs

implant – 96f10cfa6ba24c9ecd08aa6d37993fe4

File path – %SystemRoot%\System32\sipauth32.tsp

## Registry – HKLM\Software\Microsoft\Windows\CurrentVersion\Telephony\Providers

---

Source: <https://securelist.com/darkpulsar/88199/>