MythicAgents/Apollo

github.com/MythicAgents/Apollo

MythicAgents

MythicAgents/
Apollo

A .NET Framework 4.0 Windows Agent

9 Contributors
8 Issues
358 Stars
63 Forks

Apollo is a Windows agent written in C# using the 4.0 .NET Framework designed to be used in SpecterOps training offerings.

Installation

To install Apollo, you'll need Mythic installed on a remote computer. You can find installation instructions for Mythic at the Mythic project page.

From the Mythic install directory, use the following command to install Apollo as the **root** user:

```
./mythic-cli install github https://github.com/MythicAgents/Apollo.git
```

From the Mythic install directory, use the following command to install Apollo as a **non-root** user:

```
sudo -E ./mythic-cli install github
https://github.com/MythicAgents/Apollo.git
```

Once installed, restart Mythic to build a new agent.

Notable Features

- P2P Communication
- Credential Tracking and Manipulation
- Unmanged PE, .NET Assembly, and PowerShell Script Execution
- User Exploitation Suite
- SOCKSv5 Support

Commands Manual Quick Reference

Command	Syntax	Description
assembly_inject	`assembly_inject -PID [pid] -Assembly [assembly] -Arguments [args]`	Execute .NET assembly in remote process.
blockdlls	`blockdlls -EnableBlock [false]`	Block non-Microsoft signed DLLs from loading into post-ex jobs.
cat	`cat -Path [file]`	Retrieve the output of a file.
cd	`cd -Path [dir]`	Change working directory.
cp	`cp -Path [source] -Destination [destination]`	Copy a file from path to destination.
dcsync	`dcsync -Domain contoso.local [-User username -DC dc.ip]`	DCSync one or more user credentials
download	`download -Path [path] [-Host [hostname]]`	Download a file off the target system.

Command	Syntax	Description
execute_assembly	`execute_assembly -Assembly [assembly.exe] -Arguments [args]`	Execute a .NET assembly registered with `register_file`
execute_pe	`execute_pe -PE [binary.exe] -Arguments [args]`	Execute a statically compiled executable that's been registered with `register_file`
exit	`exit`	Task agent to exit.
get_injection_techniques	`get_injection_techniques`	Show currently registered injection techniques as well as the current technique.
getprivs	`getprivs`	Enable as many privileges as possible for the current access token.
ifconfig	`ifconfig`	Get Network Adapters and Interfaces
inject	`inject`	Inject a new payload into a remote process.
inline_assembly	`inline_assembly -Assembly [Assembly.exe] -Arguments [Additional Args]`	Execute a .NET assembly in the currently executing process that's been registered with `register_file`
jobkill	`jobkill [jid]`	Kill a running job in the agent.
jobs	`jobs`	List all running jobs.
keylog_inject	`keylog_inject -PID [pid]`	Inject a keylogger into a remote process.

Command	Syntax	Description
kill	`kill -PID [pid]`	Attempt to kill the process specified by `[pid]`.
link	`link`	Link to a P2P agent via SMB or TCP. Modal popup only.
load	`load command1 command2 ...`	Load new commands into the agent.
ls	`ls [-Path [path]]`	List files and folders in `[path]`. Defaults to current working directory.
make_token	`make_token`	Impersonate a user using plaintext credentials. Modal popup.
mimikatz	`mimikatz -Command [args]`	Execute Mimikatz with the specified arguments.
mkdir	`mkdir -Path [dir]`	Create a directory.
mv	`mv -Path [source] -Destination [destination]`	Move a file from source to destination. Modal popup.
net_dclist	`net_dclist [domain.local]`	List all domain controllers for the current or specified domain.
net_localgroup_member	`net_localgroup_member -Group [groupname] [-Computer [computername]]`	Retrieve membership information from a specified group on a given computer.
net_localgroup	`net_localgroup [computer]`	Retrieve local groups known by a computer. Default to localhost.

Command	Syntax	Description
net_shares	`net_shares [-Computer [computer]]`	Show shares of a remote PC.
netstat	`netstat [-Tcp -Udp -Established -Listen]`	Get TCP and UDP connections
powerpick	`powerpick -Command [command]`	Executes PowerShell in a sacrificial process.
powershell	`powershell -Command [command]`	Executes PowerShell in your currently running process.
powershell_import	`powershell_import`	Register a new .ps1 file to be used in other PowerShell jobs
ppid	`ppid -PID [pid_integer]`	Set the PPID of sacrificial jobs to the specified PID.
printspoofer	`printspoofer -Command [command]`	Execute a command in SYSTEM integrity so long as you have SeImpersonate privileges.
ps	`ps`	List process information.
psinject	`psinject -PID [pid] -Command [command]`	Executes PowerShell in the process specified by `[pid]` . Note: Currently stdout is not captured of child processes if not explicitly captured into a variable or via inline execution (such as `$(whoami)`).

Command	Syntax	Description
pth	`pth -Domain [domain] -User [username] -NTLM [ntlm_hash] [-AES128 [aes128_key] -AES256 [aes256_key] -Run [program.exe]]`	Use `mimikatz`'s pth module to spawn a process with alternate credentials.
pwd	`pwd`	Print working directory.
reg_query	`reg_query -Hive [HKCU:\	HKU:|HKLM:|HKCR:] [-Key [keyname]]`
reg_write_value	`reg_write_value -Hive [HKCU:\	HKU:|HKLM:|HKCR:] -Key [keyname] [-Name [value_name] -Value [value_value]]`
register_assembly	`register_assembly`	Register a .NET assembly with the agent to be used in .NET post-exploitation activities
register_file	`register_file`	Register a file to the agent's file cache. Used to store assemblies, executables, and PowerShell scripts.
rev2self	`rev2self`	Revert the access token to the original access token.
rm	`rm -Path [path] [-Host [hostname] -File [filename]]`	Remove a file specified by `[path]`. Alternatively, if `-File` is provided, `-Path` will be used as the directory, and `-File` will be the filename.

Command	Syntax	Description
run	`run -Executable [binary.exe] -Arguments [args]`	Runs the binary specified by `[binary.exe]` with passed arguments (if any).
sc	`sc [-Query\	-Start|-Stop|-Create|-Delete] [-Computer [computername] -DisplayName [display_name] -ServiceName [servicename] -BinPath [binpath]]`
screenshot_inject	`screenshot_inject -PID [pid] [-Interval [int] -Count [int]]`	Get a screenshot of the desktop session associated with `PID` every `Interval` seconds for `Count` screenshots.
screenshot	`screenshot`	Get a screenshot of the current screen.
set_injection_technique	`set_injection_technique [technique]`	Set the injection technique used in post-ex jobs that require injection.
shell	`shell [command]`	Run a shell command which will translate to a process being spawned with command line: `cmd.exe /S /c [command]`
shinject	`shinject`	Inject given shellcode into a specified pid. Modal popup only.
sleep	`sleep [seconds]`	Set the callback interval of the agent in seconds.

Command	Syntax	Description
socks	`socks -Port [port]`	Standup the socks server to proxy network traffic, routable via Mythic on `[port]`.
spawn	`spawn`	Spawn a new callback in the postex process specified by `spawnto_*`.
spawnto_x64	`spawnto_x64 -Application [path] -Arguments [args]`	Sets the process used in jobs requiring sacrificial processes to the specified `[path]` with arguments `[args]`.
spawnto_x86	`spawnto_x86 -Application [path] -Arguments [args]`	Sets the process used in jobs requiring sacrificial processes to the specified `[path]` with arguments `[args]`.
steal_token	`steal_token [pid]`	Attempts to steal the process's primary token specified by `[pid]` and apply it to our own session.
unlink	`unlink`	Unlink a callback linked to via the `link` command. Modal popup only.
upload	`upload`	Upload a file to a remote path on the machine. Modal popup only.
whoami	`whoami`	Report access token for local and remote operations.

Supported C2 Profiles

HTTP Profile

The HTTP profile calls back to the Mythic server over the basic, non-dynamic profile. When selecting options to be stamped into Apollo at compile time, all options are respected with the exception of those parameters relating to GET requests.

SMB Profile

Establish communications over SMB named pipes. By default, the named pipe name will be a randomly generated GUID.

TCP Profile

Establish communications over a specified network socket. Note: If unelevated, the user may receive a prompt to allow communications from the binary to occur over the network.

SOCKSv5 Support

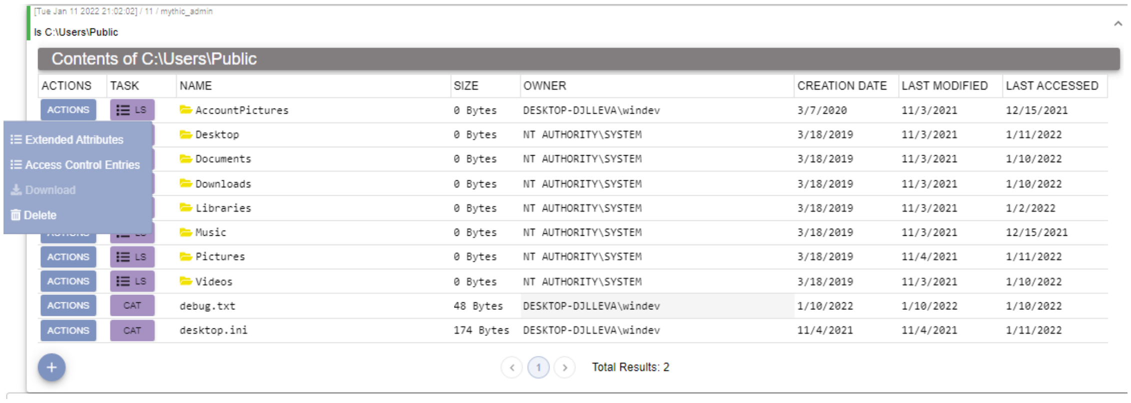
Apollo can route SOCKS traffic regardless of what other commands are compiled in. To start the socks server, issue `socks -Port [port]`. This starts a SOCKS server on the Mythic server which is `proxychains4` compatible. To stop the SOCKS proxy, navigate to the SOCKS page in the Mythic UI and terminate it.

Quality of Life Improvements

File Triage

The `ls` command reports back a wealth of information and allows operators to easily copy file paths and examine permissions of files, in addition to being able to sort and filter files. Clicking the icon under the ACLs column will show all the permissions of a file. Additionally, this hooks into Mythic's native file browser.

This shows typical ls output:



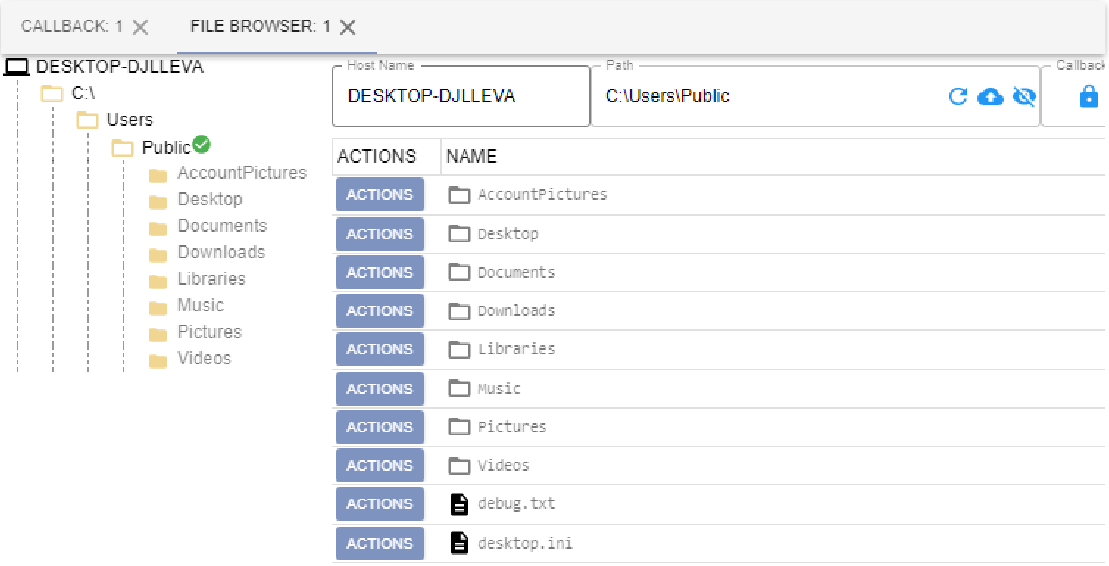
[Tue Jan 11 2022 21:02:02] / 11 / mythic_admin

ls C:\Users\Public

Contents of C:\Users\Public

ACTIONS	TASK	NAME	SIZE	OWNER	CREATION DATE	LAST MODIFIED	LAST ACCESSED
ACTIONS	LS	AccountPictures	0 Bytes	DESKTOP-DJLLEVA\windev	3/7/2020	11/3/2021	12/15/2021
		Desktop	0 Bytes	NT AUTHORITY\SYSTEM	3/18/2019	11/3/2021	1/11/2022
		Documents	0 Bytes	NT AUTHORITY\SYSTEM	3/18/2019	11/3/2021	1/10/2022
		Downloads	0 Bytes	NT AUTHORITY\SYSTEM	3/18/2019	11/3/2021	1/10/2022
		Libraries	0 Bytes	NT AUTHORITY\SYSTEM	3/18/2019	11/3/2021	1/2/2022
		Music	0 Bytes	NT AUTHORITY\SYSTEM	3/18/2019	11/3/2021	12/15/2021
ACTIONS	LS	Pictures	0 Bytes	NT AUTHORITY\SYSTEM	3/18/2019	11/4/2021	1/11/2022
ACTIONS	LS	Videos	0 Bytes	NT AUTHORITY\SYSTEM	3/18/2019	11/3/2021	1/10/2022
ACTIONS	CAT	debug.txt	48 Bytes	DESKTOP-DJLLEVA\windev	1/10/2022	1/10/2022	1/10/2022
ACTIONS	CAT	desktop.ini	174 Bytes	DESKTOP-DJLLEVA\windev	11/4/2021	11/4/2021	1/11/2022

Interfaces with Mythic's filebrowser and caches data server-side:



Host Name: DESKTOP-DJLLEVA

Path: C:\Users\Public

Callback

ACTIONS	NAME
ACTIONS	AccountPictures
ACTIONS	Desktop
ACTIONS	Documents
ACTIONS	Downloads
ACTIONS	Libraries
ACTIONS	Music
ACTIONS	Pictures
ACTIONS	Videos
ACTIONS	debug.txt
ACTIONS	desktop.ini

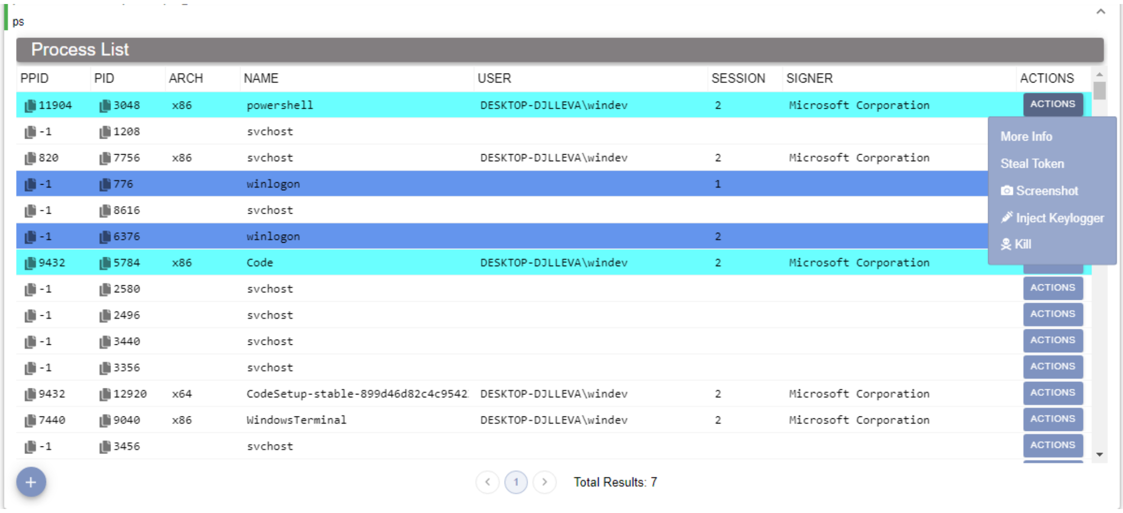
Process Listings

When issuing `ps`, additional details are retrieved such as:

- Company name of the process executable
- Description of the process executable
- Full path of the process
- Integrity level of the process
- Desktop session
- Process command line arguments

This process listing also interfaces with Mythic's builtin process browser, which allows you to see process trees more easily.

Lastly, the associated browser script will do row highlighting based on the process's name (in a one-to-one port of this script)



ps

Process List

PPID	PID	ARCH	NAME	USER	SESSION	SIGNER	ACTIONS
11904	3048	x86	powershell	DESKTOP-DJLLEVA\windev	2	Microsoft Corporation	ACTIONS
-1	1208		svchost				
820	7756	x86	svchost	DESKTOP-DJLLEVA\windev	2	Microsoft Corporation	
-1	776		winlogon		1		
-1	8616		svchost				
-1	6376		winlogon		2		
9432	5784	x86	Code	DESKTOP-DJLLEVA\windev	2	Microsoft Corporation	
-1	2580		svchost				ACTIONS
-1	2496		svchost				ACTIONS
-1	3440		svchost				ACTIONS
-1	3356		svchost				ACTIONS
9432	12920	x64	CodeSetup-stable-899d46d82c4c9542	DESKTOP-DJLLEVA\windev	2	Microsoft Corporation	ACTIONS
7440	9040	x86	WindowsTerminal	DESKTOP-DJLLEVA\windev	2	Microsoft Corporation	ACTIONS
-1	3456		svchost				ACTIONS

Total Results: 7

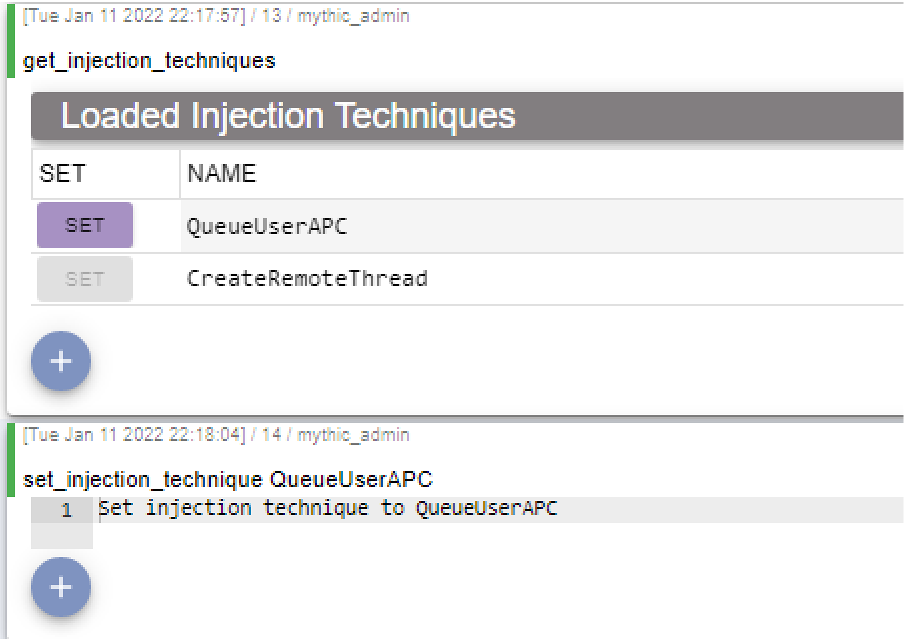
Portable Executable, Assembly, and PowerShell Script Caching

Apollo can cache files for expeditious task execution. In general, control flow follows the `register_file` command followed by whatever command you wish to execute (`execute_assembly` , `powerpick` , `execute_pe` , etc.). These files are cached client side via DPAPI encrypted AES256 blobs, preventing their signatures being exposed outside of task execution.

Dynamic Injection Techniques

The agent can change what code injection technique is in use by post-exploitation jobs that require injection through a suite of injection commands. Currently, injection techniques that are supported:

- CreateRemoteThread
- QueueUserAPC (Early Bird)



[Tue Jan 11 2022 22:17:57] / 13 / mythic_admin

get_injection_techniques

Loaded Injection Techniques

SET	NAME
SET	QueueUserAPC
SET	CreateRemoteThread

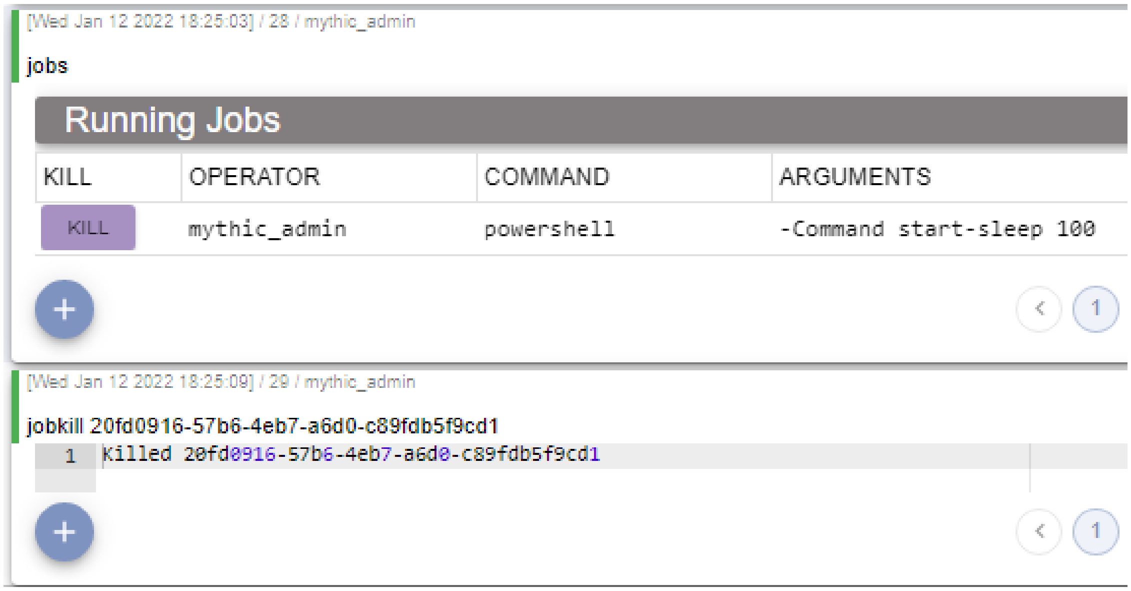
[Tue Jan 11 2022 22:18:04] / 14 / mythic_admin

set_injection_technique QueueUserAPC

```
Set injection technique to QueueUserAPC
```

Job Tracking

Agent jobs are tracked by job ID, by command, and by the arguments passed to the command so that you know what job correlates to what tasking.



```
[Wed Jan 12 2022 18:25:03] / 28 / mythic_admin
jobs
```

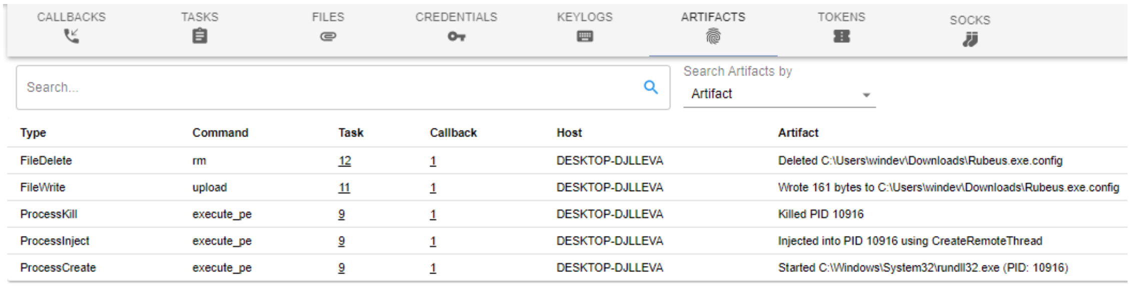
Running Jobs

KILL	OPERATOR	COMMAND	ARGUMENTS
KILL	mythic_admin	powershell	-Command start-sleep 100

```
[Wed Jan 12 2022 18:25:09] / 29 / mythic_admin
jobkill 20fd0916-57b6-4eb7-a6d0-c89fdb5f9cd1
Killed 20fd0916-57b6-4eb7-a6d0-c89fdb5f9cd1
```

Artifact Tracking

Commands that manipulate the disk, create new logons, or spawn new processes will document those changes in the Artifact Reporting page as shown below.



Search Artifacts by: Artifact

Type	Command	Task	Callback	Host	Artifact
FileDelete	rm	12	1	DESKTOP-DJLLEVA	Deleted C:\Users\windev\Downloads\Rubeus.exe.config
FileWrite	upload	11	1	DESKTOP-DJLLEVA	Wrote 161 bytes to C:\Users\windev\Downloads\Rubeus.exe.config
ProcessKill	execute_pe	9	1	DESKTOP-DJLLEVA	Killed PID 10916
ProcessInject	execute_pe	9	1	DESKTOP-DJLLEVA	Injected into PID 10916 using CreateRemoteThread
ProcessCreate	execute_pe	9	1	DESKTOP-DJLLEVA	Started C:\Windows\System32\rundll32.exe (PID: 10916)

And more!

There's a number of niceities that come with pairing an agent to Mythic - too many to list in one README. Install the agent and see for yourself!

Special Thanks

A big thanks goes to those who have contributed to the project in both major and minor ways.

- Cody Thomas, @its_a_feature_
- Calvin Hedler, @001SPARTaN

- Lee Christensen, @tifkin_
- Brandon Forbes, @reznok
- Thiago Mayllart, @thiagomayllart
- Matt Hand, @matterpreter
- Hope Walker, @IceMoonHSV
- Jack Ullrich, @winternl_t
- Elad Shamir, @elad_shamir
- Ben Turner @benpturner
- Ian Wallace @strawp
- m0rv4i @m0rv4i
- Harley Lebeau @r3dQu1nn
- Antonio Quina @st3r30byt3
- Sean Pierce @secure_sean