Institiúid Teicneolaíochta Cheatharlach

INSTITUTE of TECHNOLOGY CARLOW

At the Heart of South Leinster

WINTRE: AN ADVERSARY EMULATION TOOL

User Manual

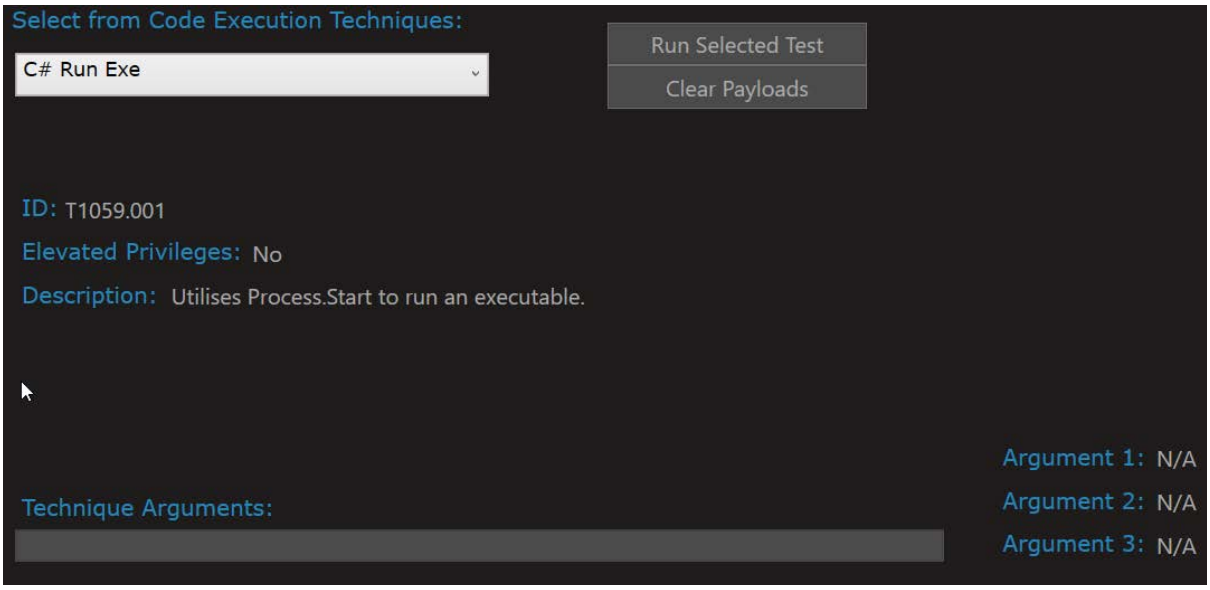
Student: Martin Earls / C00227207
Supervisor: Richard Butler

Contents

1 TECHNIQUES

1.1 RUNNING TECHNIQUES MANUALLY

By default, when WINTRE loads you will be presented with the Techniques page. From here you can manually run techniques and view any standard output that is generated. After a technique is selected the techniques, information will be displayed including its corresponding MITRE ATT&CK ID, whether or not it requires elevated privileges, a brief description, and any arguments (if applicable).



Select from Code Execution Techniques: C# Run Exe

ID: T1059.001

Elevated Privileges: No

Description: Utilises Process.Start to run an executable.

Argument 1: N/A

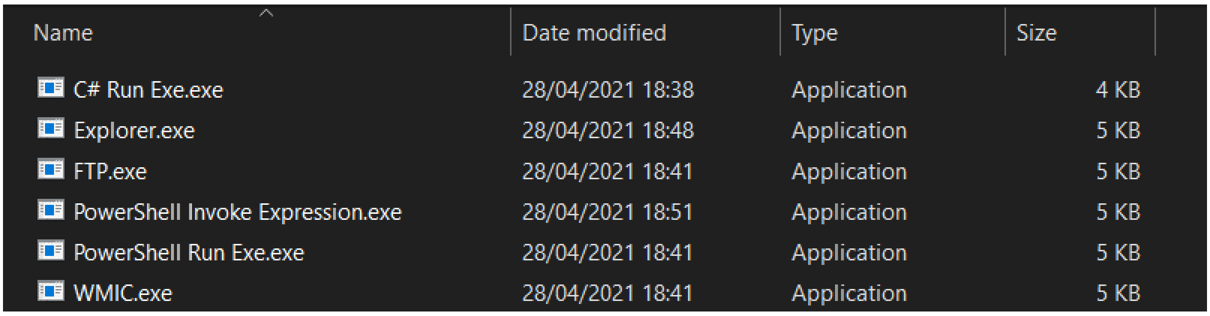
Argument 2: N/A

Argument 3: N/A

Technique Arguments:

Run Selected Test will execute the currently selected technique. **Clear Payloads** will delete the compiled executables generated from running techniques. Note it is not essential to clear the payloads, the option is simply there if needed.

Executables that are compiled from running techniques will be found in the **Payloads** folder.

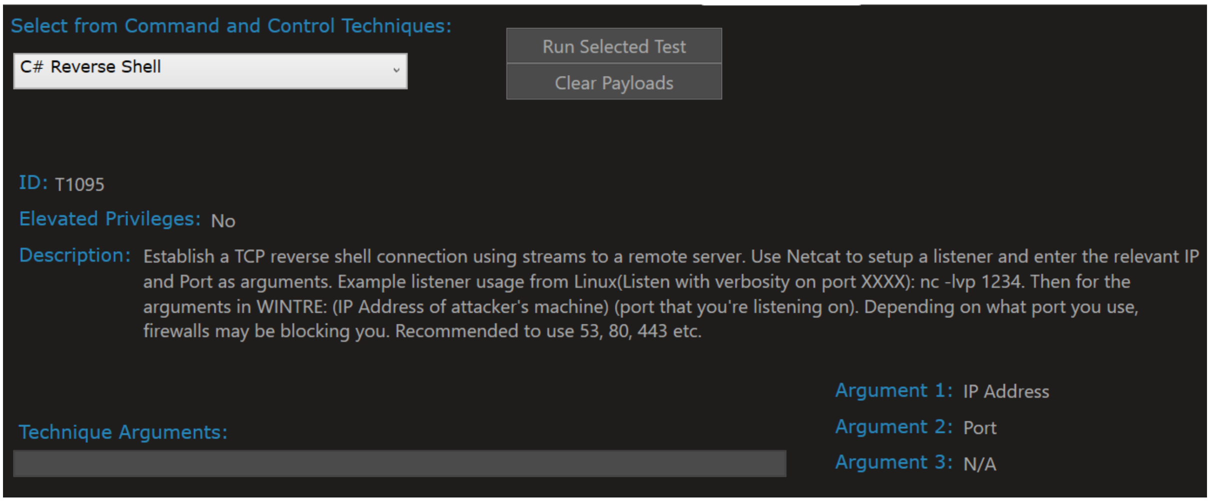


Name	Date modified	Type	Size
C# Run Exe.exe	28/04/2021 18:38	Application	4 KB
Explorer.exe	28/04/2021 18:48	Application	5 KB
FTP.exe	28/04/2021 18:41	Application	5 KB
PowerShell Invoke Expression.exe	28/04/2021 18:51	Application	5 KB
PowerShell Run Exe.exe	28/04/2021 18:41	Application	5 KB
WMIC.exe	28/04/2021 18:41	Application	5 KB

The execution of these techniques will be logged to **WINTRE-log.txt** in the same folder as WINTRE.exe for that session. If you want to make each technique more discernible from one another while parsing your logs, it is advised to wait a minute before running a new technique.

1.2 TECHNIQUES WITH ARGUMENTS

Some techniques require arguments in order to be ran. Once the technique is selected you will see the necessary arguments at the lower right hand of the techniques interface. Some arguments may be optional and will indicate so in their technique's description.



ID: T1095

Elevated Privileges: No

Description: Establish a TCP reverse shell connection using streams to a remote server. Use Netcat to setup a listener and enter the relevant IP and Port as arguments. Example listener usage from Linux(Listen with verbosity on port XXXX): nc -lvp 1234. Then for the arguments in WINTRE: (IP Address of attacker's machine) (port that you're listening on). Depending on what port you use, firewalls may be blocking you. Recommended to use 53, 80, 443 etc.

Argument 1: IP Address

Argument 2: Port

Argument 3: N/A

For example, the C# reverse shell needs additional arguments to run. Due to the nature of a reverse shell a secondary machine is required to run this technique.

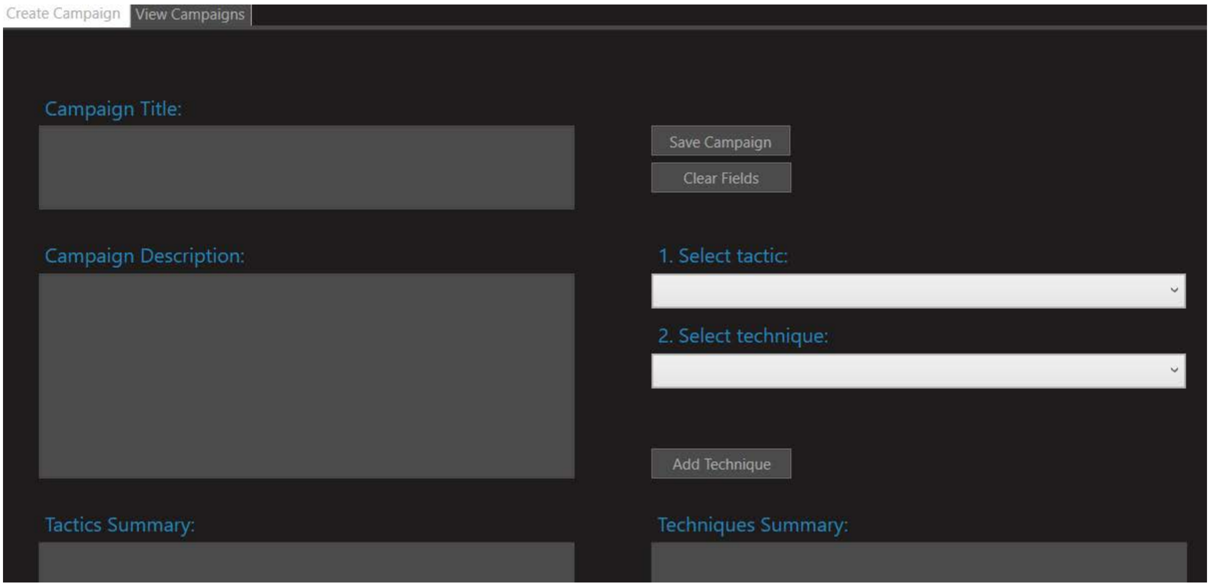
Example usage:
"Attacker's" machine at 192.168.1.1 (receiving the shell to the "victim" running WINTRE):
nc -lvp 53

You then enter the relevant arguments (192.168.1.1 53) and click **Run Selected Test**.

2 Campaigns

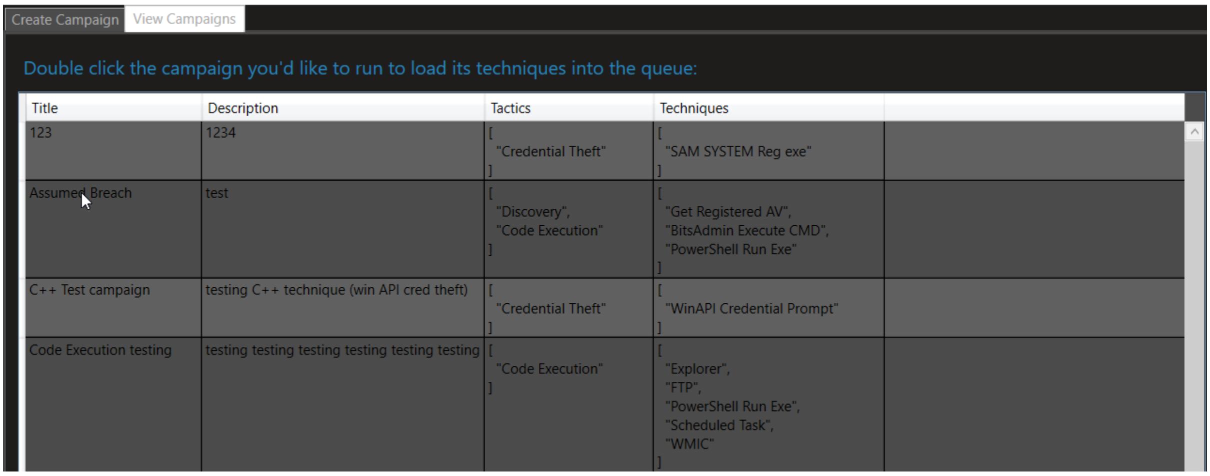
2.1 Creating a campaign

A campaign is simply a group of techniques you want to save, in order to help keep track of which techniques have been ran and allow for ease of re-testing. In order to create a campaign, enter a title, description and select your techniques based on their corresponding tactic (category of technique).



2.2 Loading a campaign

Once you've saved your campaign it will appear in the view campaigns tab where you can double click on its row to load it. Campaigns can be deleted in the **Campaigns** folder of WINTRE where campaigns are saved in JSON format.

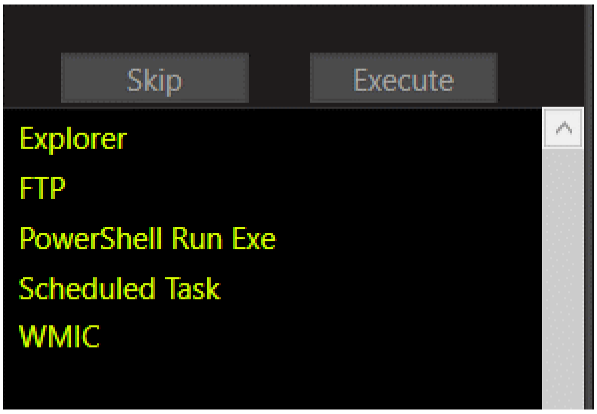


Double click the campaign you'd like to run to load its techniques into the queue:

Title	Description	Tactics	Techniques
123	1234	["Credential Theft"]	["SAM SYSTEM Reg exe"]
Assumed Breach	test	["Discovery", "Code Execution"]	["Get Registered AV", "BitsAdmin Execute CMD", "PowerShell Run Exe"]
C++ Test campaign	testing C++ technique (win API cred theft)	["Credential Theft"]	["WinAPI Credential Prompt"]
Code Execution testing	testing testing testing testing testing testing	["Code Execution"]	["Explorer", "FTP", "PowerShell Run Exe", "Scheduled Task", "WMIC"]

2.3 Running a Campaign

After loading a campaign, the techniques will appear in the queue in the main window. From here you can execute each one individually or skip any technique in the queue. Please note that more complex techniques such as reverse shells that require arguments cannot be run from the campaigns window. Trying to execute one that requires arguments will give you an error message and allow you to run or skip the remaining techniques.

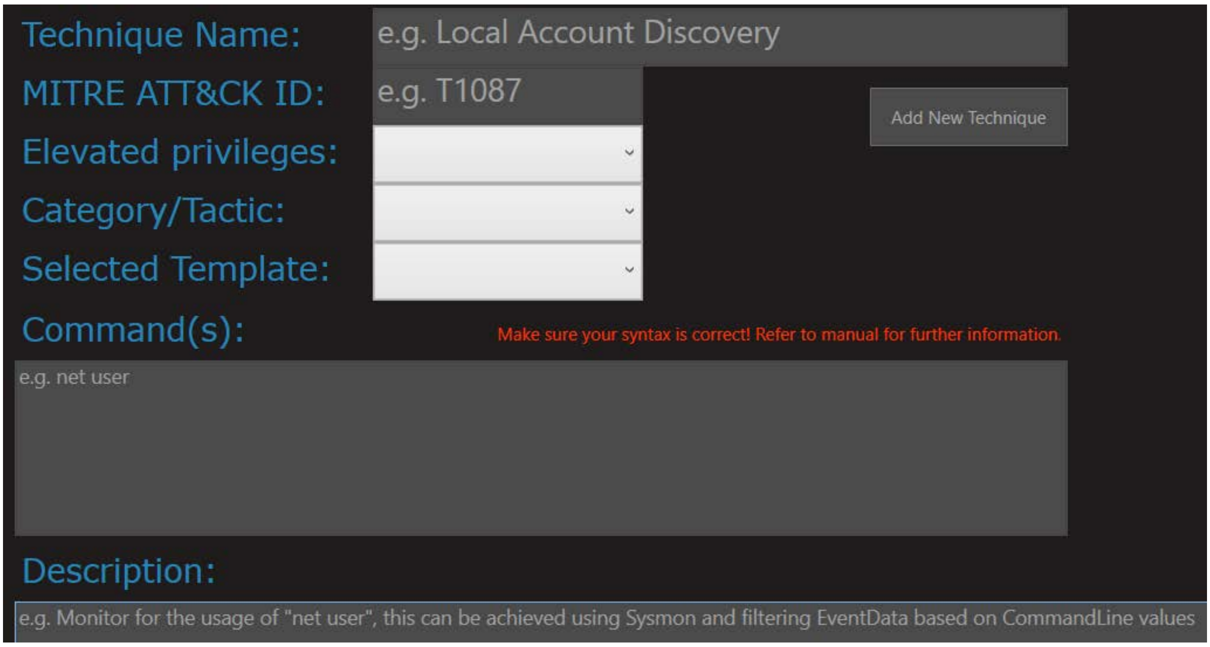


Explorer
FTP
PowerShell Run Exe
Scheduled Task
WMIC

3 CUSTOM TECHNIQUES

3.1 COMMAND LINE TECHNIQUES

WINTRE allows the creation of custom techniques, focusing on command-line techniques. A custom command of your choosing will be added to a C# source code file, compiled and executed normally as any of the built-in techniques, many of which were added to WINTRE using this interface. The Custom Techniques page contains example values. Note you can only use alphanumerical characters for the technique name. For the MITRE ATT&CK ID, please refer to the MITRE ATT&CK Framework based on what your command is performing.



Technique Name: e.g. Local Account Discovery

MITRE ATT&CK ID: e.g. T1087

Elevated privileges:

Category/Tactic:

Selected Template:

Command(s):

Make sure your syntax is correct! Refer to manual for further information.

e.g. net user

Description:

e.g. Monitor for the usage of "net user", this can be achieved using Sysmon and filtering EventData based on CommandLine values

When deciding what command to use for your new technique, make sure to test it with either Command Prompt or PowerShell wrapping them in the same format that WINTRE will execute them with. Special characters will be escaped and should translate without issue.

Command Prompt:
cmd /c "**your command**"

PowerShell:
powershell -command "**your command**"

If you make a mistake or want to delete a technique it can be deleted directly from the **TTPs** folder.

3.2 Beyond Command Line

Techniques are loaded directly based on their source code, if you want to create more powerful custom techniques using C# or C++ to utilise the Windows API you can program that technique as an individual program and then add its source code to the TTPs folder in its relevant tactic (category of technique) folder. You can also add a **.json** file in the same folder based on the structure accepted by WINTRE. Note any **.cs** or **.cpp** files added here will be loaded on the techniques page in WINTRE.

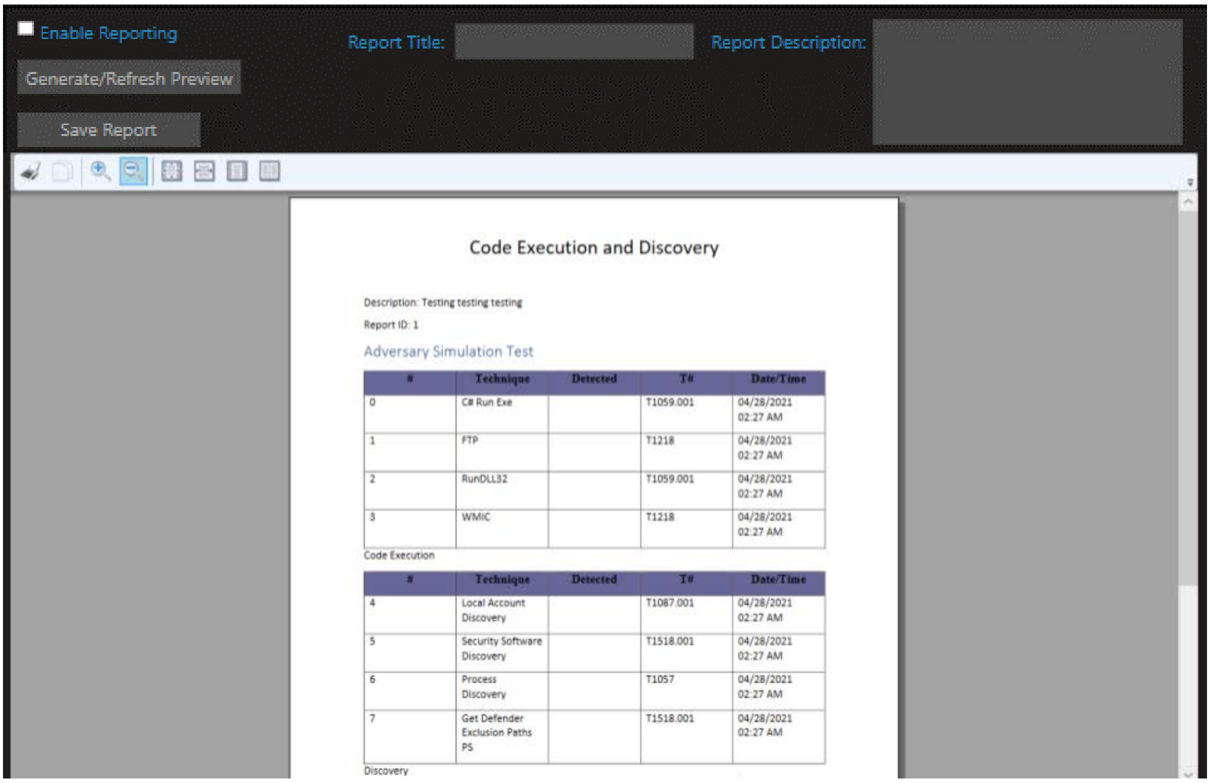
```
{
 "name": "C# Run Exe",
 "ID": "T1059.001",
 "elevated": "No",
 "tactic": "Code Execution",
 "template": "PowerShell",
 "commands": "",
 "desc": "Utilises Process.Start to run an executable.",
 "hasArgs": "false",
 "arg1": "N/A",
 "arg2": "N/A",
 "arg3": "N/A",
 "isCPP": "false"
}
```

JSON structure for techniques. Note "commands" is optional.

WINTRE does not currently support complex non-command-line techniques, i.e., if you tried adding a large project's multiple source code files they would not compile correctly and likely cause exceptions.

4 Reports

Reports allows the user to optionally generate a Microsoft Word report based on the techniques that have ran. The report can act as a template saving an analyst the trouble of creating the tables themselves.



Code Execution and Discovery

Description: Testing testing testing

Report ID: 1

Adversary Simulation Test

#	Technique	Detected	T#	Date/Time
0	C# Run Exe		T1059.001	04/28/2021 02:27 AM
1	FTP		T1218	04/28/2021 02:27 AM
2	RunDLL32		T1059.001	04/28/2021 02:27 AM
3	WMIC		T1218	04/28/2021 02:27 AM

Code Execution

#	Technique	Detected	T#	Date/Time
4	Local Account Discovery		T1087.001	04/28/2021 02:27 AM
5	Security Software Discovery		T1518.001	04/28/2021 02:27 AM
6	Process Discovery		T1057	04/28/2021 02:27 AM
7	Get Defender Exclusion Paths PS		T1518.001	04/28/2021 02:27 AM

Discovery

To create a report:

1. Enter a title and description.
2. Check Enable Reporting.
3. Run techniques either from campaigns or from the techniques page.
4. Generate a preview to see what your report currently looks like.
5. Click Save Report when done.

Note: Reporting once enabled, will be enabled for that session until the application closes. Also the report is actually being saved in .docx form as you run techniques, if you have a power outage for example you may still be able to recover the temporary report file by checking the **Reports** folder.