Department of Information Engineering and Computer Science

Bachelor's Degree in Computer Science

Final Dissertation Attachment

BAS Tools Framework

Supervisor

Prof. Silvio Ranise

Student

Matteo Bregola

Co-supervisors

Dott. Pietro De Matteis

Dott. Matteo Rizzi

Dr. Salvatore Manfredi

Academic year 2023/2024

Contents

Abstract

This document is attached to the thesis. It contains two elements:

1. A description of the 164 elements that comprise the framework presented in the thesis' Chapter 4, organized by category.

2. The caldera case study, the results of which were analyzed in Chapter 5 of the thesis.

This file does not address these two elements or offer any considerations; it should only be evaluated in the context of the thesis.

1 Framework Elements

1.1 Architecture

GUI

Name: GUI

Reference: A1

Description: The tool offers a graphical user interface (GUI).

Advantages:

- It increases the overall tool usability,
- The referenced paper highlights how visual components enhance cyber situation awareness in red teaming.

Notes:

1. Yuen et al. [12] highlight visualisation's importance in achieving awareness while performing cyber red teaming and how different visualisation types are aligned with different tasks and infrastructures.

CLI

Name: CLI

Reference: A2

Description: The tool offers a command line interface (CLI).

Advanteges:

- CLI can be suited for red team experts to quicken operations and can be used to repeat operations more straightforwardly.

API

Name: API

Reference: A3

Description: The tool offers an application programming interface (API).

Advantages:

- Permits to improve communication for software developers,
- Enables to integrate the tool into other software,
- Allows to streamline operations,
- It can be used to improve and personalize the tool functionalities, adapting them to the user's needs.

On-Premise Deployment

Name: On-Premise Deployment

Reference: A4

Description: The tool can be installed directly on the client system.

Advantages:

- Results can be more relevant compared to the SaaS approach,
- It is more focused on customization compared to a SaaS approach [2],
- It offers no dependency on internet connections,
- It can grant more privacy and security compared to a SaaS approach.

Disadvantages:

- It may face scalability challenges,
- Usually, this approach needs a higher initial investment than SaaS [2],
- It requires manual maintenance.

Notes:

1. The upsides and drawbacks listed are described as a comparison of the other deployment features for the *BAS* use case. Thus, it doesn't comprehend all the elements of comparison that can be analysed when studying different deployment approaches. It is also worth mentioning that a unique product could offer a hybrid solution.

Cloud-Based Deployment

Name: Cloud-Based Deployment

Reference: A5

Description: The tool can be hosted and run by a cloud service provider (e.g., AWS, Azure, Google Cloud...).

Advantages:

- It typically permits a fast setup [9, 7],
- It usually requires less maintenance compared to an on-premise solution [9],
- It is easily scalable [8],
- It can support pay-per-use models [8].

Disadvantages:

- It can be less performant than an on-premise solution [3],
- It can be less reliable than an on-premise solution,
- It can cause security and privacy concerns [3],
- It may increase the bandwidth costs [3].

SaaS

Name: SaaS

Reference: A6

Description: The tool can be offered as a service over an internet connection.

Advantages:

- It permits to satisfy on-demand requests (scalable) and pay-per-use models [2, 10],
- It permits an easy and quick configuration [2],
- It has almost no maintenance, allowing to cut costs [2, 10],
- Usually it has a lower initial investment than on-premise solutions [2, 10],
- It Avoids problems related to software installations (requirements, compatibility..).

Disadvantages:

- It may cause security and privacy concerns [2],
- It suffers from availability concerns [2],
- It suffers from reliability concerns [10],
- It is usually less user-personalized and elastic compared to the cloud and on-premise deployment [2],
- It can be less efficient than cloud and on-premise deployment.

Direct Deployment

Name: Direct Deployment

Reference: A7

Description: The entire simulation software is installed and run on the machine to test.

Advantages:

- It can offer an easier configuration because it doesn't require to connect the attacker with the victim,
- It can have high control over the machine to which it is installed and, therefore, high testing capabilities.

Disadvantages:

- It cannot comprehend the attacks related to lateral movement,
- Installing the software directly on all the needed machines of the infrastructure can require a lot of time
- It is difficult to keep updated the software in all the machines,
- It can consume too many resources to install the whole software on the machine and some of them may not support it,
- It requires manual work to have a global view of the infrastructure security.

Notes:

1. The precursors of *BAS Tools* were libraries like the Atomic Red Team[1] used to perform a single test on the machine. These libraries have experienced a process of "automation" that led to creating tools that can run attacks more easily and from other machines. For example, Invoke-Atomic[2] is a PowerShell module that could be (with precautions) consid-

[1] https://github.com/redcanaryco/atomic-red-team
[2] https://github.com/redcanaryco/invoke-atomicredteam

ered a "direct deployment" *BAS Tool* because it performs attacks on the system simulating the payloads used by an attacker. Even if this is far from the conception of the *BAS Tool* that was earlier provided in the work, this feature makes the reader conscious that software running on a machine acting as controller and attacker could be remotely considered a *BAS Tool*.

Containerized Deployment

Name: Containerized Deployment

Reference: A8

Description: The tool can be deployed with a container solution like a docker image.

Advantages:

- It is easily scalable and portable,
- It permits an easy maintenance.

Disadvantages:

- It may not provide all the functionalities of the other solutions,
- It can avoid problems related to deploying the software on different types of hosts and solve a possible problem of dependencies (see note 1),
- It requires the adoption of the container solution,
- It can cause more overhead than an on-premise solution,

Notes:

1. For example, MITRE Caldera v5[3] installation requires as a precondition to have Python 3.8+ and NodeJS installed on the testers' devices.

Deployment Platforms

Name: Deployment Platforms

Reference: A9

Description: List operating systems supporting the tool deployment.

Notes:

1. This feature and the next one are different from the others because they require specifying a list of values instead of the presence and absence of the feature.

---

[3]https://github.com/mitre/caldera

Target Platforms

Name: Target Platforms

Reference: A10

Description: List of operating systems, platforms and servers that can be the target of the tool's attack simulation.

Notes:

1. See note 1 of A9.

Agent-Based

Name: Agent-Based

Reference: A11

Description: The tool supports an agent-based approach for the simulation. An "agent" is a software program that runs on the "victim" machine and connects to the main software running the tool.

Advantages:

- It permits an in-depth control of the target system[4],
- It permits adapting to a network with an unstable connection; even if the victim losses temporarily the connection with the central host, it can recover it without losing the performed action data (see footnote of the previous point),
- It grants flexibility by allowing the selection of which specific area of the organization to deploy a specific test [3].

Disadvantages:

- It may require a software installation (see note 1),
- It can be less scalable than an agentless approach and, therefore, provide less coverage of the system security,
- It is usually more costly than an agentless approach if the price is based on the number of agents and the user needs to deploy many agents.

Notes:

1. Usually, like for MITRE Caldera, the agent software is an executable that is downloaded from the server of the tool but it may require manual installation.
2. There are multiple ways to deploy command and control channels like HTTP, HTTPS, and TCP. This framework doesn't analyse in detail which kinds of solutions the tool supports to maintain a high-level analysis.

---

[4]Palo Alto Cloud Security: https://www.paloaltonetworks.com/cyberpedia/what-is-the-difference-between-agent-based-and-agentless-security. Last assessed: 16/08/2024

Agentless

Name: Agentless

Reference: A12

Description: The tool supports an agentless approach. It can perform simulations on different machines without running an "agent" software on them.

Advantages:

- It doesn't require to perform manually the configuration of the
- Easier to maintain rather than agent-based approaches
- Permits to test devices that cannot support agents,
- Easy to scale (see footnote on the previous page),
- Less impact on the machines (see footnote on the previous page),

Disadvantages:

- Limited testing capabilities both for the absence of software installed on the machine and for possible network restrictions[5].

Virtual Agents

Name: Virtual Agents

Reference: A13

Description: The tool offers the possibility to create a "virtual agent" that simulates a specific machine configuration in the system without actually deploying it.

Advantages:

- It allows the simulation to be performed in a safe environment, avoiding possible damage to the production.

Disadvantages:

- It cannot be used to simulate accurately the results of the real system,
- It could be difficult to configure

Notes:

1. This feature would permit the creation of a cyber-range that mimics the system infrastructure without using external software.

---

[5]Tenable Article published on April 2023: https://www.tenable.com/blog/agents-vs-agentless-which-solution-is-right-for-your-public-cloud-environment. Last assessed on 16/08/2024

Security Requirements

Name: Security Requirements

Reference: A14

Description: The host of the software doesn't need to change its security requirements to install the tool and run it.

Advantages:

- A software that necessitates minimal modification to the system configuration is preferable since it permits more realistic results.

Notes:

1. This feature is related to the requirements of the host to use the software, not the requirements to run the simulations on the targets
2. This feature could also be considered a *Comparison Element* because it may be interesting to evaluate not only if a tool requires changing the security posture but also by how much.

General Requirements

Name: General Requirements

Reference: A15

Description: The host of the software doesn't need other software or system modifications to install the tool and run it.

Advantages:

- A software that necessitates minimal modification to the system configuration is preferable since it permits more realistic results.

Notes:

1. This feature is related to the requirements of the host to use the software, not the requirements to run the simulations on the targets.
2. This feature could also be considered a *Comparison Element* because it may be interesting to evaluate not only if a tool requires changing the system but also by how much.

1.2 Organization

Ability

Name: Ability

Reference: B1

Description: The tool offers a list of pre-defined abilities that can be run on one or multiple target systems (see note 1).

Advantages:

- Permits the user to have a more in-depth understanding of what the operation is doing thanks to the associated payload and information (see note 2),

- Since an ability has a smaller payload than a whole simulation should be easier to configure (or be created) based on the user's needs,
- Having multiple abilities allows the user to create an operation by pipelining them easily.

Notes:

1. An *Ability* contains one or multiple commands with a specific "simple" objective. An example of an *Ability* is to avoid storing the commands' logs in the bash history. This objective could be obtained by executing this payload: "$HOME/.bash_history && unset HISTFILE" (Figure 1.1 partially reports how this ability is represented in MITRE CALDERA v4.2.0).

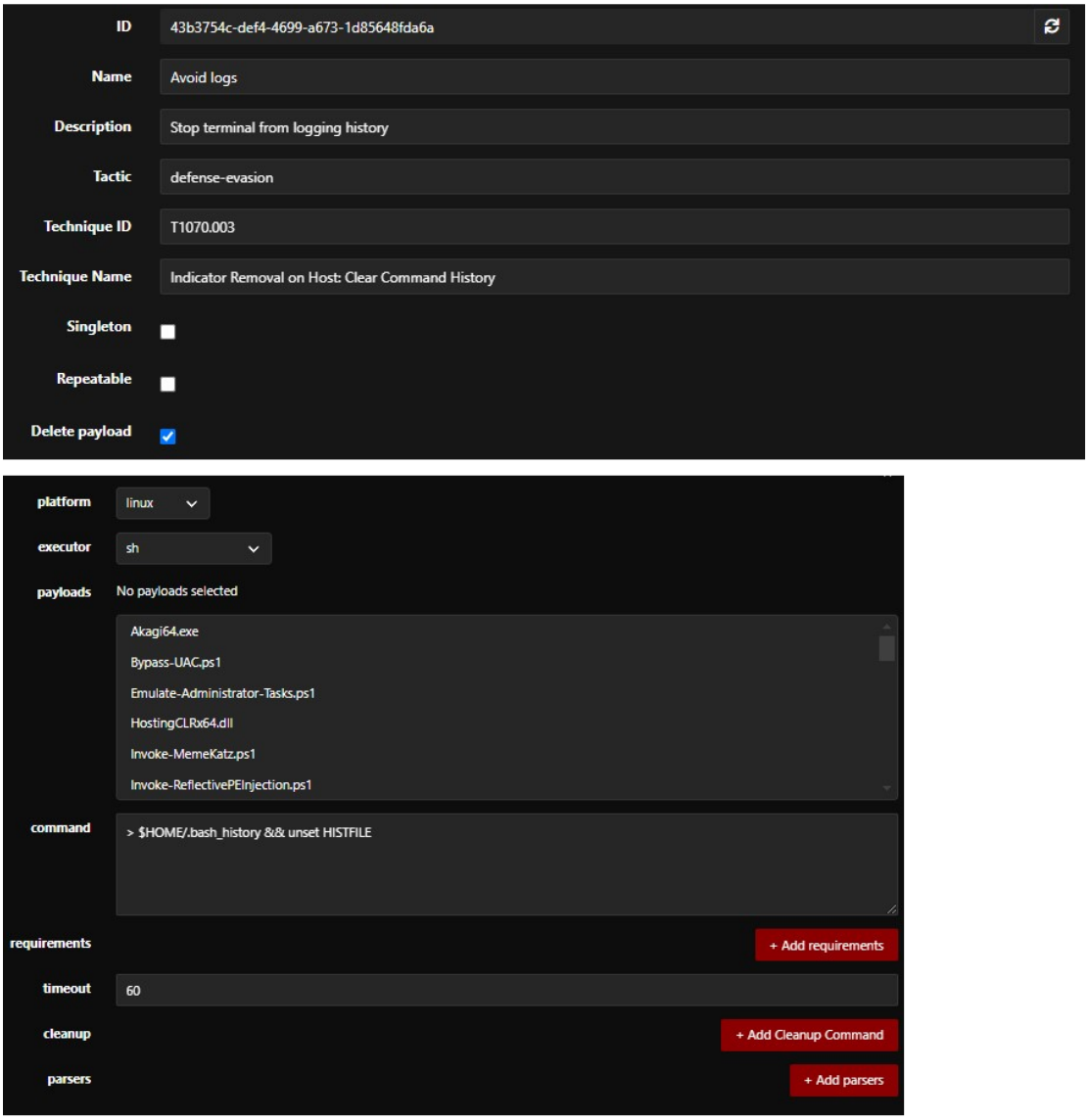

Figure 1.1: Mitre Caldera "Avoid Logs" Ability.

2. The list of possible information associated with a single ability is described in the *Information and Filtering* subsection.

Ability Configuration

Name: Ability Configuration

Reference: B2

Description: The tool permits personalizing an ability by changing pre-defined parameters.

Advantages:

- It permits configuring an ability based on the specific system or objective; it is especially useful if the tool is used by a trained red team that can personalize the abilities to their needs,
- It permits to compare the result of the same ability with different configurations,
- It avoids creating multiple similar abilities with different configurations.

Custom Ability

Name: Custom Ability

Reference: B3

Description: The tool permits creating and executing on-fly an ability; this ability can be built from scratch or by modifying an existing one

Advantages:

- It can be useful to create a custom ability to cover techniques that the tool doesn't provide,
- It can be useful to have the possibility to have custom abilities to pipeline tasks that are not directly related to an attack operation but to the general simulation control (see note 1).

Notes:

1. These types of abilities parallel to the simulation can be very different from creating and sending an email with the output of the previous ability to enabling a specific AV solution.

Custom Ability Saving

Name: Custom Ability Saving

Reference: B4

Description: The tool permits permanently saving and reusing a custom ability.

Advantages:

- It avoids recreating the same custom ability each time, simplifying the creation of custom operations.

Import Ability

Name: Import Ability

Reference: B5

Description: The tool permits importing an ability.

Advantages:

- It permits rapid exchange of abilities within the organization or communities.

Operation

Name: Operation

Reference: B6

Description: The tool can run a set of abilities to perform an operation on one or multiple target systems (see note 1).

Advantages:

- It is useful to collect abilities with a similar objective to test the system readiness against a specific tactic,
- It is fundamental to connect the single ability results and information gathered with each other, empowering the single abilities itself (see note 2),
- It is fundamental to perform a multi-step attack that simulates both a general adversary behaviour and a specific APT,

Notes:

1. In this work, there is a distinction between *Operation* and a *Simulation* based on the objective of the action. An *Objective* is a group of similar abilities with a specific goal, like discovering information about the system. *Simulation* has a broader view; it involves different operations to simulate an attacker's behaviour and usually aims to cover different parts of the Kill Chain rather than a particular objective.
2. Some abilities may not be effective or work at all if they do not have the information that another ability has gathered. For this reason, an ability output can usually be used as input of another one, miming the actual behaviour of an adversary, which uses a progressive approach. Let's take as an example the ability "Discovery Adversary" from MITRE Caldera, remembering the mapping between Caldera's *Adversary* and this framework's *Operation* presented in paragraph "Organization". Figure 1.2 shows a set of abilities that aims to discover information about the system. Many of these abilities output are parsed into facts (in the picture are the ones with a key in the "Unlocks" column) used to perform other abilities (in the picture, the one that has a lock in the "Require column"). In particular, "Find user processes" requires collecting the user name.

Ordering	Name	Tactic	Technique	Executors	Requires	Unlocks	Payload	Cleanup	
1	Identify active user	discovery	System Owner/User Discovery						
2	Find local users	discovery	Account Discovery: Local Account						
3	Identify local users	discovery	Account Discovery: Local Account						
4	Snag broadcast IP	discovery	System Network Configuration Discovery						
5	Find user processes	discovery	Process Discovery						
6	View admin shares	discovery	Network Share Discovery						
7	Discover domain controller	discovery	Remote System Discovery						
8	Discover antivirus programs	discovery	Software Discovery: Security Software Discovery						
9	Permission Groups Discovery	discovery	Permission Groups Discovery: Local Groups						
10	Identify Firewalls	discovery	Software Discovery: Security Software Discovery						
11	Discover Mail Server	discovery	Remote System Discovery						
12	Get Chrome Bookmarks	discovery	Browser Bookmark Discovery						

Figure 1.2: Mitre Caldera "Discovery" Adversary.

Operation Configuration

Name: Operation Configuration

Reference: B7

Description: The tool permits personalizing an operation by changing pre-defined parameters or performing other operations (see note 1).

Advantages:

- It permits configuring the operation based on the specific system or objective; it is especially useful if the tool is used by a trained red team that can personalize the operations to their needs,
- It permits comparing the result of the same operation with different configurations,
- It avoids creating multiple similar operations with different configurations.

Notes:

1. There can be many possible modifications, like changing the abilities' order and modality of execution.

Custom Operation

Name: Custom Operation

Reference: B8

Description: The tool permits creating and running an operation on the fly by pipelining different abilities or modifying an existing one.

Advantages:

- It permits the user to create a personalized set of abilities, including the abilities it has created into an operation.

Custom Operation Saving

Name: Custom Operation Saving

Reference: B9

Description: The tool permits permanently saving and reusing a custom operation.

Advantages:

- It avoids recreating each time the same custom operation.

Import Operation

Name: Import Operation

Reference: B10

Description: The tool permits importing an operation.

Advantages:

- It permits rapid exchange of operations within the organization or communities.

1.3 Attack

APT

Name: APT

Reference: C1

Description: The tool permits emulating a known Advanced Persistent Threat's (APT) attack path without manually configuring all the activities to simulate it.

Advantages:

- It is important to test the readiness of a system against the actions of a real attacker following exactly its attack pattern,
- It can be useful if the system has already been attacked or is known to be a possible target of a specific APT (see note 2).

Notes:

1. More features related to the simulation of APTs are present in the *Information and Filtering* subsection.
2. Some APTs target specific sectors or nations. Therefore, checking the system's readiness against them can be useful.

Group of APTs

Name: Group of APTs

Reference: C2

Description: The tool permits emulating multiple known Advanced Persistent Threats (APTs) attack paths without manually configuring all the activities to simulate them.

Advantages:

- Some APTs target specific sectors or nations, making it useful to assess readiness against them. Furthermore, certain APTs (e.g., APT19, APT20, APT22) are believed to be linked to the same government. If a system is a potential target of that government, it is prudent to evaluate readiness against all associated APTs.

Notes:

1. See the advantages of *APT* in C1.

Lateral Movement

Name: Lateral Movement

Reference: C3

Description: The tool permits performing simulations in which the number of compromised targets increases by performing lateral movement from the initial breach point.

Advantages:

- It grants a more realistic simulation, miming the actual behaviour of the attackers (see note 1).

Notes:

1. This is a fundamental feature for a *BAS* solution since one of the technology's core purposes is to emulate an attacker, and their typical behaviour consists of moving between the target's network.

Multiple Targets

Name: Multiple Targets

Reference: C4

Description: The tool permits performing a simulation on multiple targets simultaneously.

Advantages:

- It permits to speed up the time required to perform the simulations,
- It permits a collective overview of the results of a specific group of targets without manually aggregating the results of the single targets' simulations.

Multiple Objectives

Name: Multiple Objectives

Reference: C5

Description: The activities span different categories of attacks, not only specific ones (see note 1).

Advantages:

- It is required to have a more complete overview of the system's security.

Notes:

1. For example, the tool should offer activities that belong to all the tactics of MITRE ATT&CKand not only to a subset of them.
2. Some tools are developed to simulate only specific attacks. Flightsim, for example, is focused only on "safely generate malicious network traffic pattern"[6].

Pre-Compromise

Name: Pre-Compromise

Reference: C6

Description: The tool permits the execution of activities belonging to the pre-compromise phase (see notes 1 and 2).

Advantages:

[6]https://github.com/alphasoc/flightsim

- It grants a more complete overview of the system's security, It can partially reduce the need for other types of risk assessment (see note 3).

Notes:

1. Accordingly to what is explained in paragraph "Limitations", with "pre-compromise", we mean the actions performed by a malicious actor before taking control of the system.
2. Even though most tools focus primarily on activities that the attacker can perform after the initial foothold, some have capabilities on the pre-compromise domain. For example, OpenBAS [7]can conduct spearfishing attacks through mail and SMS. Furthermore, the tool permits the creation of social engineering attacks by sending fake media articles that affect the organisation's reputation.
3. Although *BAS Tools* are not intended to replace activities as penetration testing, incorporating pre-compromise activities can be useful for gaining general insights into the system's security in this aspect.

Reusable Information

Name: Reusable Information

Reference: C7

Description: The tool can collect data obtained through one activity and reuse it in other activities (e.g., passwords, usernames...).

Advantages:

- It can be useful to mimic an attacker more realistically: the activities it performs are not independent, but each piece of information collected is used in other actions.

Notes:

1. This mechanism of reusing information can be complex and elaborated. For example, MITRE Caldera uses parsers to extract "facts" from the output of an ability. Facts can also have a relationship with each other, and facts can be a requirement for other abilities.

System Detection

Name: System Detection

Reference: C8

Description: The tool can automatically recognise if the activities are supported by the targets, avoiding running incompatible activities.

Advantages:

- It reduces the simulation's execution time,
- It increases the results' pertinence.

Notes:

[7]https://docs.openbas.io/latest/usage/injects

1. In the *Information and Filtering* section, the features *Target OS* and *Requirements* are listed, which users can utilize to check the compatibility of activities with the target. It would be beneficial for the software to be able to perform this compatibility check automatically.

Activities Filtering

Name: Activities Filtering

Reference: C9

Description: The tool allows filtering among possible activities to determine which ones to include in the simulation.

Advantages:

- It permits speeding up the choice of activities to perform,
- It can help in creating groups of abilities with similar characteristics.

Notes:

1. The term *Activity* comprehends both *Action* and *Operation* as described in the *Information and Filtering* subsection.
2. A variety of possible attributes that can be used for filtering are described in the "filtering" features in the *Information and Filtering* subsection.
3. This feature is a prerequisite to satisfy the "filtering" features of the *Information and Filtering* subsection.

Activities Combined Filtering

Name: Activities Combined Filtering

Reference: C10

Description: The tool allows filtering among possible activities, combining different attributes, to determine which ones to include in the simulation.

Advantages:

- It permits to speed up the choice of activities to perform,
- It can help in creating groups of abilities with similar characteristics.

Notes:

1. This feature can be considered as a specification of the previous one. While the latter generally check if the tool provides a filtering capability, this one more specifically controls if the filtering methods can be combined logically. For example, the tool should permit filtering based on the ones created by Dave, which hasn't been updated recently.
2. See the notes on the previous feature.

Target Filtering

Name: Target Filtering

Reference: C11

Description: The tool allows filtering among possible targets to determine which ones to include in the simulation.

Advantages:

- It permits to speed up the choice of targets to test,
- It can help check the security for targets with similar characteristics.

Planners

Name: Planners

Reference: C12

Description: The tool permits the usage or creation of planners capable of deciding in which order and how to execute the activities.

Advantages:

- The order of execution of activities may change significantly the simulation's result,
- They permit the addition of logic in the simulation execution.

Notes:

1. The planner has the objective to decide not only the logic behind the execution order but also the modalities; for example, a specific planner may require the user's approval before executing each activity, while another may execute them automatically.
2. Mitre Caldera offers the *Atomic Planner* as default; it also offers other options like the "Batch" and "Bucket", and it also permits the creation of custom ones.

Prioritisation

Name: Prioritisation

Reference: C13

Description: The tool permits running the activities chosen for the simulation in a prioritized order.

Advantages:

- If there is a need to perform a complex simulation involving numerous activities, it may be beneficial to first test the most critical ones, analyse the partial results (if the tool supports live reporting), and then decide how to proceed with the simulation based on those results.

Notes:

1. The priority order can be chosen by different attributes, like the ones described in the "Information and Filtering" sub-category.

1.4 Results

Manual Report

Name: Manual Report

Reference: D1

Description: The tool permits the user to generate a report after the simulation ends.

Advantages:

- It prevents the user from needing to manually analyse the result of the simulation and create a summary or presentation, The reports, in general, facilitate the understanding of the results.

Notes:

1. There could be many features related to the report itself and what elements it contains. For example, reports may be downloadable or customizable, containing details like who performed the simulation, the security trend, etc. However, adding these aspects would complicate the framework with less important information.

Automatic Report

Name: Automatic Report

Reference: D2

Description: The tool automatically generates one or more reports after the simulation.

Advantages:

- It prevents the user from manually generating a report each time,
- The reports, in general, facilitate the understanding of the results,
- It can be used to organize work with pipelines where many simulations are scheduled, and the reports are analysed together.

Automatic Report Sending

Name: Automatic Report Sending

Reference: D3

Description: The tool can be configured to automatically generate reports and send them to specific users thanks to mail, ticketing systems or other integrations.

Advantages:

- It increases the overall efficiency of the team by automating useless passages,
- Permit to integrate *BAS* in a standardized workflow.

Role Based Reports

Name: Role Based Reports

Reference: D4

Description: The tool generates reports with different styles based on the receivers' role.

Advantages:

- It permits aligning the tool's usage with the company's organization and personnel's subdivision,
- It avoids creating ad-hoc reports with different purposes and levels of detail.

Notes:

1. For example, it may be useful to generate a detailed report for the cybersecurity team while a more general one for the managerial teams of the company.

Analytics

Name: Analytics

Reference: D5

Description: The tool returns the analytics of the simulation with a parsable format or offers an API to do use them.

Advantages:

- It permits analysing in detail the results,
- A parsable format can be exploited for creating reusable tool extensions (see note 3).

Notes:

1. For example, the tool may generate files with the results in a format like CSV, JSON or XML.
2. This feature, or the *Results Filtering* one, is a prerequisite to satisfy the "results" features of the *Information and Filtering* subsection.
3. For example, it would be possible to create a Python module that takes the results file as input and performs standard operations each time without the need to apply the same filters each time.

Result Filtering

Name: Result Filtering

Reference: D6

Description: The tool's GUI permits filtering the results.

Advantages:

- It permits the restriction of the range of the results,
- It permits detailed analysis of the results, focusing on specific elements.

Notes:

1. See note 2 of *Analytics*.

Framework-Based Results

Name: Framework-Based Results

Reference: D7

Description: The tool maps the results to a specific framework or convention.

Advantages:

- It simplifies the understanding of the results,
- It provides a better organization of the results.

Notes:

1. Many tools provide a mapping with MITRE ATT&CKanalysing the system's efficiency against each attack tactic.

System Response

Name: System Response

Reference: D8

Description: The report categorises in different levels how the system responded to the various activities of the simulation.

Advantages:

- It gives a clear and schematic way of understanding the system's readiness against the various vulnerabilities.

Notes:

1. For example, NetSPI's *BAS*[8] response categories are: *Logged*, *Detected*, *Altered*, *Responded* and *Prevented*. Each category can have one value among the following: *Full*, *Partial* and *None*.
2. This feature is essential to aid the security team and follow the *BAS* objective of filling the gap between the red team and the blue team.

---

[8]https://www.netspi.com/breach-and-attack-simulation/

System Vulnerability Levels

Name: System Response

Reference: D9

Description: The tool offers a quantitative evaluation of how much the system is vulnerable to each activity performed during the simulation (see note 1).

Advantages:

- It gives a clear and schematic way of understanding the system's readiness against the various vulnerabilities.

Notes:

1. Vulnerability is intended as the product of two elements:
   - How well the system responded to the attack (see *System Response*)
   - The probability of success (or the likelihood) of that attack

   An approach similar to this is present in Cymulate's *BAS Tool*[9].
2. The evaluation can use a numeric scale.
3. Each target of the simulation can have a different vulnerability level for the same activity; with this feature, the framework analyses the overall performance of the system, averaging all the targets. In the following feature, the value is divided by targets.

Targets Vulnerability Levels

Name: Targets Vulnerability Levels

Reference: D10

Description: The tool offers a quantitative evaluation of how much each target is vulnerable to each activity performed during the simulation.

Advantages:

- It gives a clear and schematic way of understanding the readiness of each target against the various vulnerabilities,
- It can be used to verify which targets are more vulnerable in the system.

Notes:

1. See notes 1 and 2 of "System Vulnerability Levels"
2. To have this feature the tool should support the possibility of running a simulation on multiple targets concurrently (see *Multiple Targets* in *Attack* section) or performing lateral movement (see *Lateral Movement* in *Attack* section).

---

[9] https://cymulate.com/breach-and-attack-simulation/

System Overall Vulnerability Level

Name: System Overall Vulnerability Level

Reference: D11

Description: The tool offers a quantitative evaluation of how much the system is vulnerable based on the simulation result.

Advantages:

- It is a key performance indicator(KPI) of the system's security performance against the specific simulation.

Notes:

1. See notes 1 and 2 of *System Vulnerability Levels*.
2. This value is related to the result of a specific simulation, and not many of them like the *General Vulnerability Score* in the *Others* category.
3. The difference between this feature and *System Vulnerability Levels* is that this describes the average of the various vulnerability levels of the system instead of having a value for each activity.

Targets Overall Vulnerability Level

Name: Targets Overall Vulnerability Level

Reference: D12

Description: The tool offers a quantitative evaluation of how much each target is vulnerable based on the simulation result.

Advantages:

- It is a KPI of the system security performance of each target against the specific simulation.
- It can be used to verify which targets are more vulnerable in the system.

Notes:

1. See notes 1 and 2 of *System Vulnerability Levels* and note 2 of *Targets Vulnerability level*.
2. The difference between this feature and *Targets Vulnerability Levels* is that this describes the average of the various vulnerability levels within a target instead of having a value for each activity.

System Risk Levels

Name: System Risk Levels

Reference: D13

Description: The tool delivers a quantitative evaluation of the risk level to the system resulting from the activities simulated (see note 1).

Advantages:

- It gives a clear and schematic way of understanding the system's risk against the various vulnerabilities tested in the simulation.

Notes:

1. Risk is known as the product between likelihood and impact. While the likelihood could be easily evaluated (see *System Vulnerability Level Feature*), the impact strictly depends on the business and stakeholders. For this reason, even if theoretically it would be possible for a tool to offer this feature, it is practically very difficult to develop it objectively.
2. The evaluation can use a numeric scale.
3. Each simulation target can have a different vulnerability level for the same activity; with this feature, the framework analyses the system's overall performance, averaging all the targets. In the following feature, the value is divided by targets.
4. The tool proposed by Ferraz [1] permits users to specify a weight for each technique that is then used to evaluate the "overall severity" of the technique

---

Targets Risk Levels

Name: Targets Risk Levels

Reference: D14

Description: The tool offers a quantitative evaluation of how high the risk for each target resulting from the activities simulated.

Advantages:

- It gives a clear and schematic way of understanding the potential risk of each target resulting from the various vulnerabilities,
- It can be used to verify which targets are more at risk in the system.

Notes:

1. See notes 1,2, and 3 of *System Risk Levels*, and note 2 of *Targets Vulnerability level.*

System Overall Risk Level

Name: System Overall Risk Level

Reference: D15

Description: The tool offers a quantitative evaluation of how high the risk is for the system based on the simulation result.

Advantages:

- It is a KPI of the system security performance against the specific simulation.

Notes:

1. See notes 1,2, and 3 of *System Risk Levels.*

Targets Overall Risk Level

Name: Targets Overall Risk Level

Reference: D16

Description: The tool offers a quantitative evaluation of how high the risk is for each target based on the simulation result.

Advantages:

- It is a KPI of the system security performance of each target against the specific simulation.
- It can be used to verify which targets represent a higher risk in the system.

Notes:

1. See notes 1,2, and 3 of *System Risk Levels* and note 2 of *Targets Vulnerability level*.
2. The difference between this feature and *Targets Risk Levels* is that this describes the average of the various risk levels within a target instead of having a value for each activity.

Vulnerabilities Visualization

Name: Vulnerabilities Visualization

Reference: D17

Description: The tool permits to graphically visualize the system's vulnerabilities in a meaningful way.

Advantages:

- It simplifies the understanding of the results,
- It can provide a better organization of the results.

Notes:

1. This can be done with different types of charts and can contain different information like the vulnerability level, the system response and the other described earlier.
2. See notes 1 and 2 of *Attack Path Visualization*.

Threat Grouping

Name: Threat Grouping

Reference: D18

Description: The tool groups the threats discovered during the simulation in a specific way.

Advantages:

- It simplifies the understanding of the results.

Attack Path Visualization

Name: Attack Path Visualization

Reference: D19

Description: The tool enables graphical visualization of the attack path of the simulation, displaying the sequence of steps performed.

Advantages:

- It simplifies the understanding of the results,
- it can provide a better organization of the results.

Notes:

1. Similarly to what has been said in *Framework-Based Results*, many solutions map their results to MITRE ATT&CK, showing a matrix that can be read to understand the sequence of steps performed by the simulation. Figure 1.3, taken from the Mitre Attack Navigator by selecting the APT18, contains the activities performed by the APT and can be read from left to right in order to understand the attack path. Different tools provide an in-house implementation similar to this.

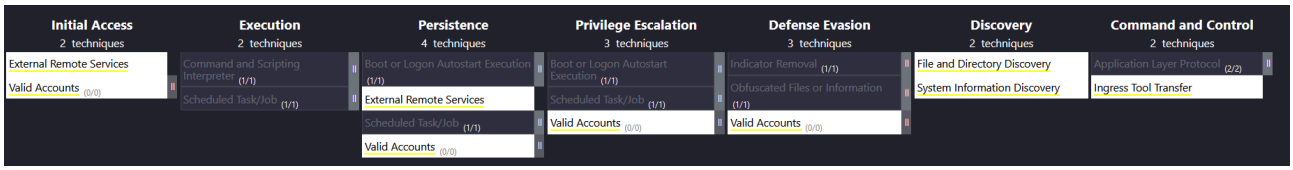

Figure 1.3: MITRE ATT&CKNavigator APT18.

2. The difference between this feature and *Vulnerabilities Visualization* one is that the latter focuses more on the sequence of the activities while the former on the results.

Discovered Targets

Name: Discovered Targets

Reference: D20

Description: If the simulation involved lateral movement, the tool provides the list of discovered targets from the initial breach point in which the attack has started.

Advantages:

- It is useful to evaluate how much the network is segmented, a fundamental aspect of the system's overall security.
- It can be used as a KPI of system security.

Compromised Targets

Name: Compromised Targets

Reference: D21

Description: If the simulation involved lateral movement, the tool provides the list of discovered targets that permitted the simulation to run activities on their systems.

Advantages:

- It is useful to evaluate how much the network is segmented, a fundamental aspect of the system's overall security.
- It can be used as a KPI of system security.

Lateral Movement Visualization

Name: Lateral Movement Visualization

Reference: D22

Description: If the simulation involved lateral movement, the tool visually represents how the attacker performed it.

Advantages:

- It simplifies the understanding of how lateral movement can be performed within the system.

Notes:

1. XM Cyber platform[10] has a very effective visualization of lateral movement in which the system is divided into groups, and it permits showing with an animation how the attacker moved inside the network and what targets were leveraged to reach other ones.

Benchmark

Name: Benchmark

Reference: D23

Description: The report provides a benchmark of customer readiness in comparison with the other competitors within the same industry.

Advantages:

- It gives a more general view of the overall system's security and an element of comparison,
- It can be used as a KPI of system security.

Notes:

1. Cymulate's *BAS Tool*[11] should have this feature of offering their customer a comparison with each other. Vectr accomplishes this feature using *Industry Threat Indexes* [12].

---

[10]https://xmcyber.com/platform/
[11]https://cymulate.com/breach-and-attack-simulation/
[12]https://docs.vectr.io/user/threat-sim-indexes/

Mitigations Insights

Name: Mitigations Insights

Reference: D24

Description: The result contains suggestions on how to fix the discovered vulnerabilities and security problems.

Advantages:

- It significantly simplifies and speeds up the work of the blue team.

Notes:

1. Each solution provides different mitigation insights with different levels of precision; they may provide external references on how to solve the problem, suggest mitigation based on specific EDR software and many others.
2. The feature *Automatic Mitigation* in *Others* section verifies if the tool is automatically capable of applying a mitigation to the vulnerabilities discovered.
3. Goldberg [5] states that "the better tool provides actionable steps to close off specific attack scenarios".

Comparison

Name: Comparison

Reference: D25

Description: The tool permits to perform a comparison of the simulation's result with previous ones.

Advantages:

- It can be used by the blue team in order to evaluate the effectiveness of particular patches or changes in the configuration it has added.

Integration Comparison

Name: Integration Comparison

Reference: D26

Description: The tool permits performing a comparison of the various system's security integrations effectiveness.

Advantages:

- It can be used by the blue team in order to understand which integration has a better impact on their system's security.

Notes:

1. Many solutions permit the integration of EDR, XDR or other security software and check how each software responds to the simulation. In particular, it would be helpful to differentiate the results described in the previous features based on the specific integration response (e.g., checking the difference in the system's vulnerability levels when software X is used with respect to software Y).

Live Results

Name: Live Results

Reference: D27

Description: The tool can generate results and reports during the simulation execution.

Advantages:

- It permits to have real-time reporting of the simulation performance.

Outputs

Name: Outputs

Reference: D28

Description: The tool is capable of collecting the outputs of the activities' commands and adding them as part of the results.

Advantages:

- It provides more detail about the system's response, which is useful for experts.

Notes:

1. The ability to collect outputs is fundamental in order to satisfy the feature called *Reusable Information* of the *Attack* category.

Logs

Name: Logs

Reference: D29

Description: The tool can collect the system logs related to the simulation's activities.

Advantages:

- It provides more detail about the system's response, which is useful for experts.

Notes:

1. This ability is required to offer the *System Response* feature in this section.
2. This feature is fundamental for providing information to the blue team.
3. Mapping specific system logs to distinct simulation's activities may be challenging.

1.5 Information and Filtering

Name

Name: Name

Reference: X1

Description: The tool assigns a name to the activity.

Advantages:

- It indicates the activity's functionality.

Name Filtering

Name: Name Filtering

Reference: Y1

Description: The tool permits filtering results based on the activities' names.

Advantages:

- It can be useful in order to group similar activities.

Name Results

Name: Name Results

Reference: Z1

Description: The tool permits filtering and running activities based on their name.

Advantages:

- It can be useful in order to group similar activities.

Source Code

Name: Source Code

Reference: X2

Description: The tool provides the source code of the related activity.

Advantages:

- It grants full transparency,
- It allows expert users to have a deeper understanding of the activity,
- It can help the user decide if the activity provides a useful capability.
- The user can modify it to create a custom activity.

Explanation

Name: Explanation

Reference: X3

Description: The tool associates the activity with a textual explanation.

Advantages:

- It helps to understand the activity's actions,
- It can help the user decide if the activity provides a useful capability.

Notes:

1. Even if the source code is provided, non-expert users may find it too difficult or time-consuming to understand it. Therefore, it can aid them in understanding the activity without having to analyse the source code.

Objective

Name: Objective

Reference: X4

Description: The tool describes the activity's objective.

Advantages:

- It helps to understand the activity's actions,
- It can help the user determine if the activity offers a valuable function.

Notes:

1. See *Explanation* note 1

History

Name: History

Reference: X5

Description: The tool provides the history of the activity's executions.

Advantages:

- It can be used to check whether the system has been recently tested and potentially with what results against that particular activity.

Notes:

1. This history could also contain various elements like the results of each execution, the user that performed it and many others.

Number of Runs

Name: Number of Runs

Reference: X6

Description: The tool provides the number of times that the activity has been used.

Advantages:

- It can be useful to track and test the activities that have been less used in precedent simulations,
- It can be useful to check if there is a correlation between the number of times a particular activity has been tested and the system response permitting also to evaluate if the *BAS* functionality is leveraged effectively.

Number of Runs Filtering

Name: Number of Runs Filtering

Reference: Y6

Description: The tool permits filtering and running activities based on how many times they have been tested.

Advantages:

- It can be useful to track and test the activities that have been less used in precedent simulations,

---

Number of Runs Results

Name: Number of Runs Results

Reference: Z6

Description: The tool permits filtering results based on how many times the activities have been tested.

Advantages:

- It can be useful to check if there is a correlation between the number of times a particular activity has been tested and the system response permitting also to evaluate if the *BAS* functionality is leveraged effectively.

Framework/CTI Reference

Name: Framework/CTI Reference

Reference: X7

Description: The tool provides an external reference to a framework or CTI report that has discovered, presented or described the vulnerability associated with the activity (see note 1).

Advantages:

- It can be used by red team experts to gain a deeper understanding of the vulnerability,
- It can be used to check if the system is secure against specific new vulnerabilities,
- It gives the user a clearer understanding of what possible attacks the tool is covering (see note 2),
- It provides a common terminology and categorization, aligned with the overall cybersecurity community.

Notes:

1. The term "references" used in this feature is very general, they range from more general CTI reports to more specific "frameworks" like Cyber Kill Chain, MITRE ATT&CKand Common Vulnerabilities and Exposures (CVE)[13]. Among them, many solutions offer a mapping to MITRE ATT&CKor CVE.
2. Offering a mapping between the tool's abilities and threat information frameworks permits a broader overview of which and how many threats can be simulated. Different solutions leverage as their selling points their MITRE ATT&CK"coverage".
3. NetSPI's *BAS* solution has a "reference" section in which there is a series of more general links related to the specific activity.

Framework/CTI Filtering

Name: Framework/CTI Filtering

Reference: Y7

Description: The tool permits filtering and running activities based on frameworks or CTI information they are related to.

Advantages:

- It is useful to restrict the range of testing,
- It permits analysing a specific subsection of the framework, focusing on specific types of vulnerabilities.

Notes:

1. This feature considers two different capabilities: the first one is to select only the activities related to one specific framework, for example selecting only activities mapped to MITRE ATT&CKinstead of CVE. The second one refers to the ability to filter activities inside that specific framework. MITRE ATT&CK, for example, categorises techniques into tactics, and it is useful to filter based on them.

Framework/CTI Results

Name: Framework/CTI Results

Reference: Z7

Description: The tool permits filtering results based on the activities' framework or CTI mapping.

Advantages:

- It is useful to check the system's readiness for the various categories of the framework.

[13]https://cve.mitre.org/

Notes:

1. Frameworks like MITRE ATT&CKand Cyber Kill Chain categorize the different attack techniques in groups, defining their general objective. It can be very useful to analyse the system's result for each category. Let's suppose that the simulation discovers a high vulnerability in the Cyber Kill Chain's *Exploitation* category. This may be a more serious problem than a bad result in the *Reconnaissance* phase.

Target OS

Name: Target OS

Reference: X8

Description: The tool indicates which operating systems the activity supports.

Advantages:

- It avoids performing activities that won't work due to operating system incompatibility.

Target OS Filtering

Name: Target OS Filtering

Reference: Y8

Description: The tool permits filtering and running activities based on the operating systems they support.

Advantages:

- It avoids performing activities that won't work due to operating system incompatibility,
- It can be used to check the vulnerabilities that are common to all operating systems.

Target OS Results

Name: Target OS Results

Reference: Z8

Description: The tool permits filtering and running activities based on the operating systems they support.

Advantages:

- It can be used to check which type of operating systems are more vulnerable and what vulnerabilities are common for all of them.

Creation Date

Name: Creation Date

Reference: X9

Description: The tool reports the activity's creation date.

Advantages:

- It is beneficial for determining the activity's age, with the added possibility of discovering the duration required to patch a new vulnerability since its introduction.

Creation Date Filtering

Name: Creation Date Filtering

Reference: Y9

Description: The tool permits filtering and running activities based on their creation date.

Advantages:

- It permits simulating new activities or checking the readiness against the old ones.

Creation Date Results

Name: Creation Date Results

Reference: Z9

Description: The tool permits filtering results based on the activities' creation date.

Advantages:

- It permits analyzing the relationship between the time of creation and the defence's results.

Creator

Name: Creator

Reference: X10

Description: The tool reports the activity's creator.

Advantages:

- It simplifies the grouping of activities.
- It permits examining the personnel's performance.

Creator Filtering

Name: Creator Filtering

Reference: Y10

Description: The tool permits filtering and running activities based on their creator.

Advantages:

- It simplifies the grouping of activities.

Creator Results

Name: Creator Results

Reference: Z10

Description: The tool permits filtering results based on the activities' creator.

Advantages:

- It permits examining the personnel's performance.

Update

Name: Update

Reference: X11

Description: The tool reports the activity's last update date.

Advantages:

- It can help determine whether a particular activity is current or if it has been altered since the last execution, and in this case, it can be useful to re-run it.

Update Filtering

Name: Update Filtering

Reference: Y11

Description: The tool permits filtering and running activities based on their last update.

Advantages:

- It permits simulating activities based on their update time.

Update Results

Name: Update Results

Reference: Z11

Description: The tool permits filtering results based on the activities' last update.

Advantages:

- It permits analyzing the relationship between the activities' update time and the defence's result.

Requirements

Name: Requirements

Reference: X12

Description: The tool reports the activity's requirements.

Advantages:

- An activity can have different types of requirements; for example, it may need to change the security configurations of the target system, it can require specific software to be installed or to have collected some specific information before running.

Requirements Filtering

Name: Requirements Filtering

Reference: Y12

Description: The tool permits filtering and running activities based on the requirements they need.

Advantages:

- It avoids performing activities that won't work due to some unsatisfied requirement.

Notes:

1. This feature encompasses both positive and negative filtering. The latter is beneficial for excluding a specific set of features from the simulation if the system or target does not meet the requirements. The former can be utilized for specific operations, such as evaluating the outcome of simulating all actions necessitating disabling the antivirus.

Requirements Results

Name: Requirements Results

Reference: Z12

Description: The tool permits filtering results based on the activities' requirements.

Advantages:

- Filtering activities based on their requirements can be used to check if it is possible to provide remediation by leveraging on the requirements rather than on the activity itself.

APT Correlation

Name: APT Correlation

Reference: X13

Description: The tool provides a list of APTs that applied that activity.

Advantages:

- It increases the importance of the activity, as it has been utilized by attackers rather than being specifically created to test a particular action ad-hoc.

APT Correlation Filtering

Name: APT Correlation Filtering

Reference: Y13

Description: The tool permits filtering and running activities based on their association with an APT.

Advantages:

- It increases the importance of the activity, as it has been utilized by attackers rather than being specifically created to test a particular action ad-hoc.

Notes:

1. This feature is different from *APT* in the *Attack* section; the latter one aims to check if the tool permits emulating a complete attack vector, while the current one instead verifies if the user can filter the features based on their mapping to an APT.

APT Correlation Results

Name: APT Correlation Results

Reference: Z13

Description: The tool permits filtering results based on the activities' possible mapping to APTs.

Advantages:

- It informs the user about the most likely attackers for their systems.

Sector Correlation

Name: Sector Correlation

Reference: X14

Description: The tool offers a list of industry sectors that have been victims of that activity.

Advantages:

- It allows the choice of activities used against the same sector of the user.

Notes:

1. It can be difficult to map an activity directly to a target sector since the same activity is often used in different attacks attacking different targets. However, some sectors are victims of a particular subset of activities (operations in particular). It could be interesting to test the "most common" activities for the user's sector.

Sector Correlation Filtering

Name: Sector Correlation Filtering

Reference: Y14

Description: The tool permits filtering and running activities based on their possible association with a specific industry sector.

Advantages:

- It can be used to narrow down the number of possible activities to simulate and focus on the ones that are particularly relevant for the user,

---

Sector Correlation Results

Name: Sector Correlation Filtering

Reference: Z14

Description: The tool permits filtering results based on the activities' possible association with a specific industry sector.

Advantages:

- It can be used to filter the results only of the activities related to the same industry as the user to indicate how the system performs on the most "probable' ones (see note 1).

Notes:

1. This should be used only as an indication since it is very difficult to map the activities to a sector in a general way, and evaluating the probability of being attacked by a certain activity is difficult to evaluate.

Nation/Region Correlation

Name: Nation/Region Correlation

Reference: X15

Description: The tool offers a list of nations or regions that have been victims of that activity (see note 1).

Advantages:

- It allows to simulate activities that have been used against the same nation as the user's.

Notes:

1. As for *Sector Correlation*, it's difficult to map an activity to the system's location that has been used against, and the same activity can have been used against targets in different regions.

See note 1 of *Nation/Region Attacker Correlation*

Nation/Region Correlation Filtering

Name: Nation/Region Correlation Filtering

Reference: Y15

Description: The tool permits filtering and running activities based on their association with a specific target nation or region.

Advantages:

- It can be used to narrow down the number of possible activities to simulate and focus on the ones that are particularly relevant for the user.

Notes:

1. See note 1 of *Nation/Region Correlation* and note 1 of *Nation/Region Attacker Correlation.*

Nation/Region Attacker Correlation

Name: Nation/Region Attacker Correlation

Reference: X16

Description: The tool provides a list of countries or regions where the specific activity has been conducted.

Advantages:

- It can be useful if the user is aware of being targeted by a specific region or nation and wants to simulate activities associated with that area.

Notes:

1. The key difference between this feature and *Nation/Region Correlation* lies in their focus: this feature analyzes the origin of the activity (where it was performed), while the latter focuses on the target (where the activity is directed).
2. as mentioned in note 1 of *Nation/Region Correlation*, retrieving the origin of a specific activity is not straightforward.

Nation/Region Attacker Filtering

Name: Nation/Region Attacker Filtering

Reference: Y16

Description: The tool provides a list of countries or regions where the specific activity has been conducted.

Advantages:

- It can be used to narrow down the number of possible activities to simulate and focus on the ones that are particularly relevant for the user.

Notes:

1. See notes 1 and 2 of *Nation/Region Attacker Correlation*.

Tag

Name: Tag

Reference: X17

Description: The tool offers a list of tags associated with the activity or permits the user to create ones.

Advantages:

- It permits creating groups of activities with the same tag,
- It speeds up the process of selecting the activities.

Notes:

1. This feature could be divided to specify if the tool offers default tags and if the user can create tags. Nevertheless, it has been decided to create only one to avoid increasing the framework's complexity.

Tag Filtering

Name: Tag Filtering

Reference: Y17

Description: The tool permits filtering and running activities based on their tags.

Advantages:

- It speeds up the process of selecting the activities.
- It enables the execution of a simulation characterized by a specific element, as described by the tag attribute.

Tag Results

Name: Tag Results

Reference: Z17

Description: The tool permits filtering results based on the activities' tags.

Advantages:

- It permits to have an element for grouping the activities results and understanding the relationship between the type of activity and the result of the simulation.

System's Vulnerability

Name: System's Vulnerability

Reference: X18

Description: The tool reports for each activity their related system's vulnerability level based on its last evaluation (see note 1).

Advantages:

- It allows the user to understand how vulnerable the system was in the last test, providing more information to prioritize simulation activities.

Notes:

1. The concept of *vulnerability level* is described in the *System Vulnerabilities Levels* feature in the *Feature Results* section.
2. A *BAS Tool* could generate a vulnerability level that quantifies how much a specific target is vulnerable to a particular activity, the average of the results of all targets permits the definition of the system's vulnerability level related to that activity. Therefore is possible to store in the activity's information not only a vulnerability level associated with the system in general but also a level for each target. This feature and the next one analyse if the tool stores and evaluates the former, the latter is checked by the *Target's Vulnerability* and its related feature.
3. The difference between this feature and the *History* one is that the latter aims to check if the tool provides general information about the history of executions of that particular activity; instead, this one is specific. Therefore the *History* feature can be evaluated as present during the analysis even if it does not contain the vulnerability level.
4. Note the fact that the vulnerability level refers to a previous test. The tool should not assign the level before running the simulation; doing so would be out of scope and more related to vulnerability scanning.

System's Vulnerability Filtering

Name: System's Vulnerability Filtering

Reference: Y18

Description: The tool permits filtering and running activities based on their associated system's vulnerability level.

Advantages:

- • It permits the user to perform a simulation of the most vulnerable activities for their system in general.

Notes:

1. See note 1 of *System's Vulnerability*.

Target's Vulnerability

Name: Target's Vulnerability

Reference: X19

Description: The tool reports for each activity the associated vulnerability level of all system's targets based on their last evaluation.

Advantages:

- It offers the same benefits as *System Vulnerability* but provides a more precise view of how the vulnerability is distributed across the system.

Notes:

1. See notes 1, 2 and 4 of *System's Vulnerability*.

Target's Vulnerability Filtering

Name: Target's Vulnerability Filtering

Reference: Y19

Description: The tool permits filtering and running activities based on their associated targets' vulnerability levels.

Advantages:

- It permits restricting the range of the simulation to the activities with a higher vulnerability for the targets intended to be tested.

System's Risk

Name: System's Risk

Reference: X20

Description: The tool reports for each activity their related system's risk level based on its last evaluation (see note 1).

Advantages:

- It permits the user to have an idea of how high the system's risk was in the last test and have more elements to decide which activities are more important to simulate.

Notes:

1. The concept of *risk level* is described in the *System Risk Levels* feature in the *Feature Simulation* section.
2. The same reasoning pointed out for the second note of "System's Vulnerability" holds for the risk instead of the vulnerability.

System's Risk Filtering

Name: System's Risk Filtering

Reference: Y20

Description: The tool permits filtering and running activities based on their associated system's risk level.

Advantages:

- It permits the user to simulate the most dangerous activities for their system in general.

Notes:

1. See note 1 of System's Risk.

Target's Risk

Name: Target's Risk

Reference: X21

Description: The tool reports for each activity the associated risk level of all target systems based on their last evaluation.

Advantages:

- It offers the same benefits as "System Risk" but provides a more precise view of how the risk is distributed across the system.

Notes:

1. See note 1 and 2 of System's Risk.

Target's Risk Filtering

Name: Target's Risk Filtering

Reference: X22

Description: The tool permits filtering and running activities based on their associated targets' risk levels.

Advantages:

- It permits restricting the range of the simulation to the activities with a higher risk for the targets intended to be tested.

Notes:

1. See note 1 and 2 of Target's Risk.

Rollback

Name: Rollback

Reference: X22

Description: The tool reports if the activities offer a rollback that restores the system status as before the operation has been performed.

Advantages:

- It is essential to determine whether an activity is safe to perform on the production system.

Rollback Filtering

Name: Rollback Filtering

Reference: Y22

Description: The tool permits filtering and running activities based on whether they support rollback.

Advantages:

- It permits easy selection of the activities that can be executed in production from the ones that require a cyber range.

1.6 Simulation

Cleanup

Name: Cleanup

Reference: E1

Description: The tool permits performing a complete cleanup of the system after the execution of the simulation, restoring its original status.

Advantages:

- It is fundamental if the user wants to execute simulations in the production environment.

Notes:

1. The *Rollback* feature in the *Information and Filtering* section analyses the same capabilities but for the individual activities rather than for the whole simulation.

Simulation Live Info

Name: Simulation Live Info

Reference: E2

Description: The tool provides the user with real-time information about the simulation it is executing.

Advantages:

- It permits the user to understand what is happening to the system during the simulation.

Notes:

1. The tool can provide numerous details like the command it executes, the outputs, which targets are currently under examination, the elapsed execution time and many others.

Pause

Name: Pause

Reference: E3

Description: The tool permits pausing the simulation during its execution.

Advantages:

- It can be helpful to analyze the partial results and decide whether is useful to continue the simulation or not (see note 1),
- If the simulation is running on the production system and is causing excessive overload, it may be useful to block the execution and continue it later.
- It permits the user to perform actions between the simulation's activities (see note 2).

Notes:

1. The tool should have the *Live Results* features to permit real-time outcomes analysis.
2. The ability to perform actions between each activity could also be provided if the tool includes a planner that waits for the user's approval before executing each activity (see *Planners* in the *Results* section).

Override

Name: Override

Reference: E4

Description: The tool permits the user to pause the simulation and change its configurations.

Advantages:

- It can be helpful to analyze the partial results and change the configurations based on them (see note 1).

Notes:

1. The tool should have the "Live Results" features to permit real-time outcomes analysis.

Interrupt

Name: Interrupt

Reference: E5

Description: The tool permits the user to pause the simulation during its execution.

Advantages:

- It grants the user control over the simulation.

Schedule

Name: Schedule

Reference: E6

Description: The tool can automatically execute simulations without user interaction and permits the user to schedule them.

Advantages:

- It reduces the users' work.
- It allows for systematic work organisation and provides a standardized approach to evaluating system security over time.

Import Simulation

Name: Import Simulation

Reference: E6

Description: The tool permits importing a simulation's plan from a file.

Advantages:

- It enables rapid simulations' exchange and configuration using files.

1.7 Environment

Integrations

Name: Integrations

Reference: F1

Description: The tool can integrate other software.

Advantages:

- It enhances the tool's capabilities (see note 1),
- It permits to take advantage of already present software (see note 3).

Notes:

1. Many different types of software can be used to enhance *BAS* capabilities; for example, ticketing solutions can be useful for sending reports or assigning tasks to the different team members, AVs and EDRs can be integrated to permit the *BAS* to analyze their detection and reporting capabilities as described in the *Comparison* feature of *Results*.
2. The number of supported Integrations is one of the *Comparison Elements*.
3. OpenBAS[14] refers to its integrations as "Injectors" describing them as the platform backbone. These *Injectors* play a crucial role, offering a wide range of functionalities, from creating cyber ranges to integrating others *BAS Tools* like Caldera.

Integrable

Name: Integrable

Reference: F2

Description: The tool can be integrated into other software.

Advantages:

- If a user has other security software, it may be useful to integrate the *BAS* capabilities and results with them, creating a unified platform

Notes:

1. A BAS tool can be integrated into more complex software like *Security Orchestration, Automation and Response* (SOAR) solutions that manage all the security programs.
2. The number of software capable of integrating the tool is one of the *Comparison Elements*.

RBAC

Name: RBAC

Reference: F3

Description: The tool can be configured to follow a role-based access control (RBAC) mechanism.

Advantages:

- It grants a higher level of security,
- It permits different users to have different capabilities.

Notes:

1. The tool can be used by different people with different roles and belonging to different teams; it can be helpful to provide different access mechanisms aligned with these different scenarios. For example, It may be helpful to divide the blue team from the red teams and provide different access to managers recreating the user organizational structure in the tool. For example, *OpenBAS* has a section dedicated to managing human resources, offering three different sections: one for managing the "players", one for the "teams", and the last for the "organizations".

---

[14] https://docs.openbas.io/latest/usage/injectors/

Production Environment

Name: Production Environment

Reference: F4

Description: The tool can be used safely in production.

Advantages:

- It permits to have more realistic results,
- It avoids the need for creating cyber ranges.

Notes:

1. For example, Infection Monkey [15] categorises its plugins as "safe" or "not safe" based on whether they "cause any permanent system modifications that could impact security or operations".

Sector Specific

Name: Sector Specific

Reference: F5

Description: The tool has been developed specifically for a target sector.

Advantages:

- It permits having a more specific tool and simulation aligned with the use case.

Notes:

1. The SafeBreach website highlights how its architecture can benefit various sectors, particularly healthcare, finance, life sciences, and IT/OT environments
2. Ferraz [1] points out how different companies that offer purple team assessments aim to analyse which are the threat actors specific to the client's sector.
3. Some tools have been developed to test the readiness of specific types of infrastructure like ASTORIA for smart grids [11].

1.8 Others

Plugins

Name: Plugins

Reference: G1

Description: The tool permits the integration of plugins.

Advantages:

- It permits to personalize and improve the tool,
- It allows for a modular approach that keeps the main software simple and enables the user to decide which integrations he needs.

---

[15] https://techdocs.akamai.com/infection-monkey/docs/getting-started

CTI Updates

Name: CTI Updates

Reference: G2

Description: The tool informs the users about new vulnerabilities and CTI reports, indicating if the tool developed some new activities related to testing security against them.

Advantages:

- It helps users stay updated on new vulnerabilities.

Notes:

1. Filigran can integrate its OpenBAS solution with its OpenCTI platform that allows users to gather threat data from different sources, elaborating not only the way in which data is visualized but also creating relationships between them and many other capabilities. In particular, it is possible to generate a simulation for the *BAS Tool* by selecting a threat identified by the open platform[16].

Documentation

Name: Documentation

Reference: G3

Description: The tool offers documentation accessible to the users.

Advantages:

- It helps users understand how to use the software and what are its capabilities.

Interactive Tutorial

Name: Interactive Tutorial

Reference: G4

Description: The tool offers users an interactive tutorial explaining its usage and capabilities.

Advantages:

- It helps users understand how to use the software and what are its capabilities.

---

[16] https://docs.openbas.io/latest/usage/opencti_scenario/

General Vulnerability Score

Name: General Vulnerability Score

Reference: G5

Description: The tool quantitatively evaluates the overall system vulnerability based on all performed simulations.

Advantages:

- It helps in comprehending the overall security level of the system,
- It can be used as a system security KPI.

Notes:

1. This feature is strictly related to the one called *System Overall Vulnerability Level* in the *Results* section. The *General Vulnerability Score* could be evaluated as the mean of the various *System Overall Vulnerability Level* obtained in different simulations.
2. The tools on the market provide different "general" scores analysing different parameters and with various methodologies. For example, The Picus' *Security Validation Platform*[17] provides an "Overall Prevention Result" and an "Overall Detection Result" as KPIs; XM Cyber[18] has both a general "Security Rating" and many specific ones like the "Network Segmentation" score.

General Risk Score

Name: General Risk Score

Reference: G6

Description: The tool quantitatively evaluates the overall system risk level based on all performed simulations.

Advantages:

- It helps in comprehending the overall security level of the system,
- It can be used as a system security KPI.

Notes:

1. This feature is strictly related to the one called *System Overall Risk Level* in the *Results* section. The *General Risk Score* could be evaluated as the mean of the various *System Overall Risk Level* obtained in different simulations.
2. See note 2 of *General Vulnerability Score*

---

[17] https://www.picussecurity.com/security-validation-platform
[18] https://xmcyber.com/platform/

General Security Score

Name: General Security Score

Reference: G7

Description: The tool quantitatively evaluates the overall system security based on all performed simulations.

Advantages:

- It helps in comprehending the overall security level of the system,
- It can be used as a system security KPI.

Notes:

1. This feature can be considered as a more generalized version of the two previous ones.
2. See note 2 of "General Vulnerability Score".
3. Ferraz [1] has developed for its *BAS* solution an evaluation model to generate a score value.
4. Manocha et al. [6] explain why it could be a valid addition to have a general "security rating" for security testing purposes.

General Security Trend

Name: General Security Trend

Reference: G8

Description: The tool provides an analysis of the system's security trends over time and across simulations.

Advantages:

- It can be used to analyse the performance of the blue team and the effectiveness of adopted security solutions and investments,
- It can be used as a system security KPI.

Notes:

1. The trend can contain elements like the ones described in features G5 to G7.

Impact Level

Name: Impact Level

Reference: G9

Description: The tool allows the user to manually specify the impact of attack activities on the user's business.

Advantages:

- It can be used to evaluate the risk level of the different activities (see note 1),
- It permits the generation of more accurate security scores.

Notes:

1. As described in *System Risk Level* in the *Result* category, the risk is considered as the product between likelihood and impact since the impact is difficult to calculate automatically and depends on the stakeholders and business, it can be helpful to let the user evaluate its value for each activity manually.

Targets Relevance

Name: Targets Relevance

Reference: G10

Description: The tool allows the user to specify each asset's relevance in the system manually.

Advantages:

- It can be used to evaluate the risk level of the different activities (see note 2),
- It permits the generation of more accurate security scores.

Notes:

1. Not all system assets have the same importance for the business. For example, if the system has different databases, breaking into one of them could have a greater impact than the others.
2. See note 2 of *Impact Level* of this section.

Tag Creation

Name: Tag Creation

Reference: G11

Description: The tool allows users to create tags that can be added to activities or targets.

Advantages:

- It simplifies the selection of activities and targets.
- It permits the generation of more accurate security scores.

Notes:

1. In the *Information and Filtering* section, there are three tags-related features.

Target Customization

Name: Target Customization

Reference: G12

Description: The tool allows the user to customize the system's targets.

Advantages:

- It can enhance the tool's capabilities by giving more control over single assets.

Notes:

1. The targets can be customized in different ways, for example, by assigning tags to them or specifying rules of interaction.

Manual Targets Grouping

Name: Manual Targets Grouping

Reference: G13

Description: The tool permits the user to create and save groups of targets within the system.

Advantages:

- It simplifies the selection of the simulations' targets.

Notes:

1. For example, it can be useful to divide database servers from devices and workstations or create groups based on the IP address.
2. It can be particularly useful if the tool has the feature of *Multiple Targets* (present in *Attack Section*).
3. The following feature checks if this grouping can be done automatically by the tool.

Automatic Targets Grouping

Name: Automatic Targets Grouping

Reference: G14

Description: The tool is capable of automatically creating groups of targets within the system.

Advantages:

- It simplifies the selection of the simulations' targets.

Notes:

1. See notes on the previous feature.

Compliance

Name: Compliance

Reference: G15

Description: The tool is capable of validating compliance with one or more security standards.

Advantages:

- It reduces the effort of performing manual audits.

Standalone

Name: Standalone

Reference: G16

Description: The *BAS Tool* can be bought or used as a standalone software without integrating other functionalities.

Advantages:

- It can be more economical than solutions with multiple functionalities,
- It may avoid having overlapping solutions that perform the same things.

Notes:

1. Many tools like Picus[19] and XM Cyber[20] describe *BAS* as one of the various use cases of their software.

Automatic Mitigation

Name: Automatic Mitigation

Reference: G17

Description: The tool can apply mitigations for the discovered vulnerabilities.

Advantages:

- It significantly simplifies and speeds up the work of the blue team.

Notes:

1. Prelude's Detect platform[21] can apply remedies to discovered vulnerabilities by leveraging defensive integrations.
2. This feature is not strictly part of the *Breach and Attack Simulation* concept since it goes beyond the simulation of an attack and moves towards a blue team responsibility. A *BAS Tool* primary objective is not to fix discovered vulnerabilities. However, it may be valuable to note the existence of hybrid solutions with similar capabilities that also address vulnerability remediation.

1.9 Comparison Elements

Coverage

Name: Coverage

Reference: H1

Description: The tool can be evaluated based on how many activities it offers and how much these activities can cover different types of attacks.

Advantages:

[19] https://www.picussecurity.com/use-case/breach-and-attack-simulation
[20] https://xmcyber.com/use-case/breach-and-attack-simulation/
[21] https://www.preludesecurity.com/products/detect

- A tool should be as complete as possible, offering users as many activities as possible, allowing them to check different vulnerabilities. If a solution provides few activities or is restricted to a subtype of attack, then the security evaluations based on the simulations are not comprehensive.

Notes:

1. It is difficult to objectively give a metric of a tool's coverage for different reasons. Firstly, different tools may be based on different frameworks, and it would require an important effort to create a common mapping between them that permits comparison. Furthermore, It is important to take precautions when evaluating and using coverage; different articles [22] point out how many vendors treat the coverage as a check-list without performing a deep analysis. Therefore, even though tough coverage is important when comparing solutions, the client should avoid using it as an exclusive comparison element.

Effectiveness

Name: Effectiveness

Reference: H2

Description: The tool can be evaluated based on how effective the activities and simulations proposed are.

Advantages:

- Along with the number of activities proposed by the tool, it's essential to assess their efficiency and determine whether they can effectively achieve their intended objectives.

Price and ROI

Name: Price and ROI

Reference: H3

Description: The tool can be evaluated based on its cost, payment modalities, and predicted return on investment.

Advantages:

- Even though the price and ROI are general comparison elements for software, evaluating them in the *BAS* case is essential. One of the benefits of adopting a *BAS* solution is the potential reduction of cybersecurity costs; on the other hand, different types of security assessments or solutions can be chosen, so the price should be competitive with them.

---

[22] https://www.forrester.com/blogs/the-mitre-attck-framework-is-not-a-bingo-card/

Privacy

Name: Privacy

Reference: H4

Description: The tool can be evaluated based on its ability to safeguard user's privacy.

Advantages:

- Especially if the tool has access to the real production environment, it's fundamental that its privacy is aligned with the user's policies to grant their validity.

Security

Name: Security

Reference: H5

Description: The tool can be evaluated based on its security level.

Advantages:

- Especially if the tool has access to the real production environment, it must be secure since any security vulnerabilities of the tool can have a repercussion on the user system.

Customer Support

Name: Customer Support

Reference: H6

Description: The tool can be evaluated based on the customer support offered by the company or community that developed the tool.

Advantages:

- One of the benefits of *BAS Tools* is reducing the cost of read teams, allowing clients to perform security assessments to less trained experts. In this optic, it is fundamental that the vendor or the developer supports the team using the tool, especially in the initial phases of configuration and personalization of the solution.

Notes:

1. Some companies. like NetSPI [23], assign their consultants during the initial phases to assist clients in configuring the platform. Furthermore, other companies offer training solutions for their software.

---

[23] https://www.netspi.com/breach-and-attack-simulation/

Integrability

Name: Integrability

Reference: H7

Description: The tool can be evaluated based on its ability to integrate with other solutions or incorporate other software.

Advantages:

- It is important in order to understand if the tool can be used together with other solutions, reducing the problem of having multiple security software to manage independently.

Notes:

1. It can be seen as a quantitative evaluation of the features of *Integrations* and *Integrable* in the *Features Environment* category.

Operator Expertise

Name: Operator Expertise

Reference: H8

Description: The tool can be evaluated based on the level of cybersecurity expertise request by the software users.

Advantages:

- Accordingly, with what has been said in the *Customer Support* feature, *BAS Tools* are useful in order to reduce the need for a high-specialized red team; therefore, it is important to understand what level of cybersecurity training the user requires to employ the software.

Resource Usage

Name: Resource Usage

Reference: H9

Description: The tool can be evaluated based on how many resources are consumed on the system that is the target of the simulation.

Advantages:

- If the tool runs simulations directly on assets of the system and not on a cyber-range, it is fundamental that the activities are not too resource-consuming. Otherwise, they can create problems with the system's functionality.

Scalability

Name: Scalability

Reference: H10

Description: The tool can be evaluated based on how many assets it can support and how easy it is to scale this number.

Advantages:

- It is important to consider whether the tool is suitable for the system's size and can scale accordingly. Additionally, a scalable approach can be useful for testing the software's effectiveness on a subset of the system before deciding whether to extend its usage to the entire architecture.

Customization

Name: Customization

Reference: H11

Description: The tool can be evaluated based on how much it can be customizable and adapted to align the simulation capabilities with the user system and business interest.

Advantages:

- A customizable and flexible tool is more effective than a tool that does not allow the user to adapt the software to its system and need, especially because each customer can use the tool in different ways (different teams) and with different purposes.

Updated

Name: Updated

Reference: H12

Description: The tool's evaluation can be based on how frequently it's updated, from both an attacker's and technology's perspectives.

Advantages:

- *BAS tools* have the main objective of simulating an attacker's behaviour; therefore, keeping the software updated to new vulnerabilities, novel attack methodologies and the new technologies adopted by the companies is fundamental to testing effectively the systems' security.

Notes:

1. Many open-source tools or projects related to *BAS*, for example, Uber Metta[24] and ATTPwn[25], are no longer supported. Therefore, there is no community support, and they may not be compatible with new operating systems and do not include the new vulnerabilities.

---

[24]https://github.com/uber-common/metta
[25]https://github.com/Telefonica/ATTPwn

Sources of CTI

Name: Sources of CTI

Reference: H13

Description: The tool can be evaluated based on how vast and valuable is the knowledge behind the tool.

Advantages:

- A tool that provides a schematic and verifiable mapping of the vulnerabilities that they are simulating is useful to understand how realistic their testing capabilities are and how much are they extended.

Notes:

1. This feature analyses what are the sources of information and the frameworks that have been used as the basis for developing the tool and are currently used to update it.
2. For example, a tool, that provides attack simulations without providing any information about why it has been chosen or if it has been used is less reliable than a tool that maps all their vulnerabilities to MITRE ATT&CK. Similarly, a tool that bases its development on different CTI sources is more trustworthy than some software that uses only one reference or no one at all.

Tool Improvement

Name: Tool Improvement

Reference: H14

Description: The tool can be evaluated based on how feasible it is to improve the tool's capabilities by developing new features or creating plugins.

Advantages:

- A tool that allows the user to be part of the software development or to create integrations can be a valuable addition, especially for expert users. Since these software are often used by cybersecurity teams that may have in-house programs, it may be preferable to have a flexible solution.

Notes:

1. This is particularly true for open-source solutions, which usually are created with this idea of collaborative development, while it is more difficult to have proprietary software that permits the customers to improve their capabilities by themselves.

Time Required

Name: Time Required

Reference: H15

Description: The tool can be evaluated based on how much time it requires to perform the simulations.

Advantages:

- It is important to have fast testing to be efficient and execute many simulations.

Influence

Name: Influence

Reference: H16

Description: The tool can be evaluated based on how much their actions influence the system's result.

Advantages:

- The tool itself should be a means to simulate attacks in the most realistic way possible without influencing the results by how it performs the actions.

Notes:

1. Elgh [4] analyses the "noise" produced by different tools (meaning the number of logs generated) and uses this characteristic as a comparison element. The paper shows that there is a higher number of logs generated by the selected *BAS Toola* while performing a simulation rather than performing that same simulation manually. This fact may influence the system to detect the attacks more easily than it would have been for a real attacker and this is an undesired behaviour.
2. The difference between this feature and the *Effectiveness* is that the latter analyses the performance of the specific activities while the former of the software in general.

Mitigations Effectiveness

Name: Mitigations Effectiveness

Reference: H17

Description: The tool can be evaluated based on how much support it offers for fixing the vulnerabilities.

Advantages:

- Since the tool can be used both by a red team and a blue team is important to consider not only how many activities it can perform but also if it can suggest mitigations.

Notes:

1. The *Mitigations Insight* and the *Automatic Mitigation* features analyse what are the capabilities of the software in aiding the blue team, this one studies how effective these capabilities are.
2. See note 2 of *Automatic Mitigation*.

2 Caldera Case Study

2.1 Architecture

GUI (A1)
• Value: V
• Note: The tool is based on an interactive web interface that can be enhanced using VueJS.
• Reference: https://github.com/mitre/caldera.

CLI (A2)
• Value: V*
• Note: The tool is not directly a command line application, but its functionalities can be used via CLI thanks to the API (e.g., delete an Agent with: curl -H KEY:$API_KEY -X DELETE http://localhost:8888/api/rest -d 'index:agents,paw:$agent_paw').
• Reference: https://caldera.readthedocs.io/en/5.0.0/The-REST-API.html#.

API (A3)
• Value: V
• Note: Caldera has developed an REST API in Python, which was released in version 2.
• Reference: https://caldera.readthedocs.io/en/5.0.0/The-REST-API.html#.

On-Premise Deployment (A4)
• Value: V
• Reference: https://github.com/mitre/caldera.

Cloud-Based Deployment (A5)
• Value: X
• Note: Mitre Foundation doesn't provide an official solution that is deployable on the cloud, though it is possible to find some Amazon Machine Images provided by third parties.

SaaS (A6)
• Value: X

Direct Deployment (A7)
• Value: X*
• Note: Caldera uses an agent-based solution. However, installing the software on a machine and testing on the same machine can be possible. Containerized Deployment (A8)
• Value: V
• Note: The Gihtub Repository offers a Docker image.

Deployment Platforms (A9)
• Value: Linux, macOS
• Reference: https://github.com/mitre/caldera.

Target Platforms (A10)
• Value: Linux, Windows, Darwin
• Note: In addition to targeting specific operating systems, the tool can perform some operations against other services and software, such as AWS, Azure, and others.
• Reference: https://github.com/mitre/caldera.

Agent-Based (A11)
• Value: V
• Note: Caldera supports different agents that can communicate with the central server in various ways: HTTP, Github Gist, DNS tunnelling, TCP, and HTML Contact.
• Reference: https://caldera.readthedocs.io/en/5.0.0/Learning-the-terminology.html#agents.

Agentless (A12)
• Value: X

Virtual Agents (A13)
• Value: X*
• Note: The team developed a Mock plugin to add a simulated agent, but official support stopped with version 4.2.0.
• Reference: https://github.com/mitre/caldera/releases, https://caldera.readthedocs.io/en/5.0.0/Plugin-library.html#mock.

Security Requirements (A14)
• Value: V*
• Note: However, to connect Caldera with the Windows agents, it may require disabling the AV, having administrator privileges, or creating an exception for the agent software.
• Reference: https://github.com/mitre/caldera.

General Requirements (A15)
• Value: X
• Note: Requires Python, Pip3, NodeJs and a compatible browser.
• Reference: https://github.com/mitre/caldera.

2.2 Organization

Ability (B1)
• Value: V
• Reference: https://caldera.readthedocs.io/en/5.0.0/Basic-Usage.html#abilities.

Ability Configuration (B2)
• Value: V
• Note: By entering in the Edit Ability Menu.
• Reference: Ability Menu, Edit Ability.

Custom Ability (B3)
• Value: V
• Reference: Ability Menu, Create an Ability.

Custom Ability Saving (B4)
• Value: V
• Reference: Ability Menu, Edit Ability, Save.

Import Ability (B5)
• Value: X

Operation (B6)
• Value: V

• Note: As explained in the Organization section, it's possible to map Caldera's Adversaries to Operations in this framework.
• Reference: https://caldera.readthedocs.io/en/5.0.0/Learning-the-terminology.html.

Operation Configuration (B7)
• Value: V
• Reference: Adversaries Menu.

Custom Operation (B8)
• Value: V
• Reference: Adversaries Menu, New Profile.

Custom Operation Saving (B9)
• Value: V
• Reference: Adversaries Menu, Import.

Import Operation (B10)
• Value: V
• Note: It's possible to import a YAML file.
• Reference: Adversaries Menu, New Profile, Save.

2.3 Attack

APT (C1)
• Value: X

Group of APTs (C2)
• Value: X

Lateral Movement (C3)
• Value: V
• Reference: https://caldera.readthedocs.io/en/5.0.0/Lateral-Movement-Guide.html.

Multiple Targets (C4)
• Value: V
• Reference: Operation Menu, Group.

Multi Objectives (C5)
• Value: V
• Reference: https://caldera.readthedocs.io/en/5.0.0/resources.html#ability-list.

Pre-Compromise (C6)
• Value: V
• Note: There exists a plugin entirely dedicated to initial access.
• Reference: https://caldera.readthedocs.io/en/5.0.0/Initial-Access-Attacks.html.

Reusable Information (C7)
• Value: V
• Reference: https://caldera.readthedocs.io/en/5.0.0/Basic-Usage.html#facts.

System Detection (C8)

• Value: V
• Note: Each ability has an associated executor that is specific for a particular OS (if for example we try to run an operation on a Linux agent that involves an ability without Linux executor then it won't be performed).
• Reference: See Notes.

Activities Filtering (C9)
• Value: V*
• Note: This feature is evaluated as present even though Caldera doesn't exactly provide what is described. In fact, this feature should permit a user to filter activities and directly select only the ones that respect the filter, but Caldera is doing only the first part, and the user has to manually select/add all of them into an "Adversary".
• Reference: Abilities Menu.

Activities Combined Filtering (C10)
• Value: V*
• Note: See previous feature's note.
• Reference: Abilities Menu.

Target Filtering (C11)
• Value: X
• Note: It is possible to create a group of agents, but there is no possibility of directly filtering targets based on their attributes.

Planners (C12)
• Value: V
• Reference: https://caldera.readthedocs.io/en/5.0.0/Basic-Usage.html#planners.

Prioritisation (C13)
• Value: X
• Note: It is possible to manually change the order of activities or select a custom planner but not directly specify to follow a priority order.

2.4 Results

Manual Report (D1)
• Value: V
• Note: The tool offers the possibility to have both a "technical" report downloadable from the operation Menu and a more readable one by using the "Debrief" plugin.
• Reference: https://caldera.readthedocs.io/en/5.0.0/Operation-Results.html.

Automatic Report (D2)
• Value: X

Automatic Report Sending (D3)
• Value: X

Role-based Reports (D4)
• Value: X*
• Note: Even though there are two types of possible reports (the one downloadable from the operation Menu and the one from the plugin), they are not enough to state that the tool offers role-based reports.

Analytics (D5)
• Value: V
• Note: Caldera permits to download a JSON or a CSV..
• Reference: Operations Menu, Download Report.

Results Filtering (D6)
• Value: X*
• Note: It's not possible to filter the whole results of the operation, but it is possible to filter the operations information .

Framework-Based Results (D7)
• Value: V
• Note: The JSON file has different fields that refers to MITRE ATTACK.
• Reference: https://caldera.readthedocs.io/en/5.0.0/Operation-Results.html.

System Response (D8)
• Value: X*
• Note: Caldera doesn't provide a class of responses, but it can say if the command failed or succeeded (this is more related to the command performance than the defence).

System Vulnerability Levels (D9)
• Value: X

Targets Vulnerability Levels (D10)
• Value: X

System Overall Vulnerability Level (D11)
• Value: X

Targets Overall Vulnerability Level (D12)
• Value: X

System Risk Levels (D13)
• Value: X

Targets Risk Levels (D14)
• Value: X

System Overall Risk Level (D15)
• Value: X

Targets Overall Risk Level (D16)
• Value: X

Vulnerabilities Visualization (D17)
• Value: X

Threat Grouping (D18)
• Value: X

Attack Path Visualization (D19)
• Value: V
• Reference: https://caldera.readthedocs.io/en/5.0.0/Lateral-Movement-Guide.html#displaying-lateral-

movement-in-debrief.

Discovered Targets (D20)
• Value: X

Compromised Targets (D21)
• Value: X

Lateral Movement Visualization (D22)
• Value: V
• Reference: https://caldera.readthedocs.io/en/5.0.0/Lateral-Movement-Guide.html#displaying-lateral-movement-in-debrief.

Benchmark (D23)
• Value: X

Mitigation insights (D24)
• Value: X

Comparison (D25)
• Value: X

Integration Comparison (D26)
• Value: X

Live Results (D27)
• Value: V*
• Note: Caldera can show the status of the abilities that are being executed and their outputs.

Outputs (D28)
• Value: V

Logs (D29)
• Value: V

2.5 Information and Filtering

Name (X1)
• Value: V

Name Filtering (Y1)
• Value: V

Name Results (Z1)
• Value: V

Source Code (X2)
• Value: V

Explanation (X3)
• Value: V

Objective (X4)

• Value: V

History (X5)
• Value: X

Number of Runs (X6)
• Value: X

Number of Runs Filtering (Y6)
• Value: X

Number of Run Results (Z6)
• Value: X

Framework/CTI Reference (X7)
• Value: V

Framework/CTI Filtering (Y7)
• Value: V

Framework/CTI Results (Z7)
• Value: V
• Reference: https://caldera.readthedocs.io/en/5.0.0/Operation-Results.html.

Target OS (X8)
• Value: V

Target OS Filtering (Y8)
• Value: V

Target OS Results (Z8)
• Value: V
• Reference: https://caldera.readthedocs.io/en/5.0.0/Operation-Results.html.

Creation Date (X9)
• Value: X

Creation Date Filtering (Y9)
• Value: X

Creation Date Results (Z9)
• Value: X

Creator (X10)
• Value: X

Creator Filtering (Y10)
• Value: X

Creator Results (Z10)
• Value: X

Update (X11)

• Value: X

Update Filtering (Y11)
• Value: X

Update Results (Z11)
• Value: X

Requirements (X12) Note: The requirements section of the abilities is present in order to check whether the ability has the required facts to be performed. It does not check, for example, if the ability requires some particular security change or software installed.
• Reference: https://caldera.readthedocs.io/en/5.0.0/Requirements.html.

Requirements Filtering (Y12)
• Value: X

Requirements Results (Z12)
• Value: X

APT Correlation (X13)
• Value: X

APT Correlation Filtering (Y13)
• Value: X

APT Correlation Results (Z13)
• Value: X

Sector Correlation (X14)
• Value: X

Sector Correlation Filtering (Y14)
• Value: X

Sector Correlation Results (Z14)
• Value: X

Nation/Region Correlation (X15)
• Value: X

Nation/Region Correlation Filtering (Y15)
• Value: X

Nation/Region Attacker Correlation (X16)
• Value: X

Nation/Region Attacker Filtering (Y16)
• Value: X

Tag (X17)
• Value: V*
• Note: Each ability has a tag that maps it to one or multiple Mitre Attack Tattics. It's not possible however to have tags on activities or user-defined tags.

• Reference: Abilities Menu.

Tag Filtering (Y17)
• Value: V*
• Note: See Tag's note.

Tag Results (Z17)
• Value: V*
• Note: See Tag's note.

System's Vulnerability (X18)
• Value: X

System's Vulnerability Filtering (Y18)
• Value: X

Targets' Vulnerability (X19)
• Value: X

Target's Vulnerability Filtering (Y19)
• Value: X

System's Risk (X20)
• Value: X

System's Risk Filtering (Y20)
• Value: X

Targets' Risk (X21)
• Value: X

Targets' Risk Filtering (Y21)
• Value: X

Rollback (X22)
• Value: X*
• Note: It doesn't directly say if the operations can be rollbacked, but the cleanup commands are part of the abilities payload.
• Reference: https://caldera.readthedocs.io/en/5.0.0/Basic-Usage.html.

Rollback Filtering (Y22)
• Value: X

2.6 Simulation

Cleanup (E1)
• Value: V*
• Note: Only some abilities provide a cleanup command.

Simulation Live Info (E2)
• Value: V
• Reference: Operations Menu.

Pause (E3)
• Value: V
• Reference: Operations Menu.

Override (E4)
• Value: V
• Reference: Operations Menu.

Interrupt (E5)
• Value: V
• Reference: Operations Menu.

Schedule (E6)
• Value: X

Import Simulation (E7)
• Value: X
• Note: It's possible to import an adversary but not an entire operation.

2.7 Environment

Integrations (F1)
• Value: X

Integrable (F2)
• Value: V
• Note: The tool is already integrated into other tools like OpenBAS.

RBAC (F3)
• Value: V

Production Environment (F4)
• Value: X*
• Note: Many abilities can be performed in the production environment because doesn't cause harm or can be reverted; however, it may require the user to be cautious about what is done and how.

Sector Specific (F5)
• Value: X

2.8 Other

Plugins (G1)
• Value: V
• Reference: https://caldera.readthedocs.io/en/5.0.0/Learning-the-terminology.html#plugins.

CTI Updates (G2)
• Value: X

Documentation (G3)
• Value: V
• Reference: https://caldera.readthedocs.io/en/5.0.0/index.html.

Interactive Tutorial (G4)
• Value: V
• Note: The tool includes the training plugin.
• Reference: https://caldera.readthedocs.io/en/5.0.0/Learning-the-terminology.html#plugins.

General Vulnerability Score (G5)
• Value: X

General Risk Score (G6)
• Value: X

General Security Score (G7)
• Value: X

General Security Trend (G8)
• Value: X

Impact level (G9)
• Value: X

Targets Relevance (G10)
• Value: X

Tag Creation (G11)
• Value: X

Target Customization (G12)
• Value: V
• Note: It's possible to assign targets (agents) to group and change settings like the sleep timer and others.
• Reference: Agents Menu.

Manual Targets Grouping (G13)
• Value: V
• Reference: Agents Menu.

Automatic Targets Grouping (G14)
• Value: V
• Note: It automatically groups red agents from blue agents.
• Reference: Agents Menu.

Compliance (G15)
• Value: X

Standalone (G16)
• Value: V

Automatic Mitigation (G17)
• Value: X

Bibliography

[1] Ferraz Tomás Almeida. Breach and attack simulator. Master's thesis, Universidade de Coimbra, 2022.

[2] Stamatia Bibi, Dimitrios Katsaros, and Panayiotis Bozanis. Business application acquisition: On-premise or saas-based solutions? *IEEE Software*, 29(3):86–93, 2012.

[3] N. Ram Ganga Charan, S. Tirupati Rao, and Dr .P.V.S Srinivas. Deploying an application on the cloud. *International Journal of Advanced Computer Science and Applications*, 2(5), 2011.

[4] Joakim Elgh. Comparison of adversary emulation tools for reproducing behavior in cyber attacks. Master's thesis, Linköping Universitya, 2022.

[5] Daniel Goldberg. Living with decade-old vulnerabilities in datacentre software. *Network Security*, 2019, 2019.

[6] Hardik Manocha, Akash Srivastava, Chetan Verma, Ratan Gupta, and Bhavya Bansal. Security assessment rating framework for enterprises using MITRE ATTCK matrix, 2021.

[7] Franklin Magalhães Ribeiro, Tarcísio da Rocha, Joanna C. S. Santos, and Edward David Moreno. A model-driven solution for automatic software deployment in the cloud. In *Information Technology: New Generations*, pages 591–601. Springer International Publishing, 2016.

[8] Marco Scavuzzo. A distributed file system over heterogeneous saas storage platforms. In *2014 16th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing*, pages 417–421, 2014.

[9] Diomidis Spinellis. Developing in the cloud. *IEEE Software*, 31(2):41–43, 2014.

[10] R. Vidhyalakshmi and Vikas Kumar. Design comparison of traditional application and saas. In *2014 International Conference on Computing for Sustainable Global Development (INDIACom)*, pages 541–544, 2014.

[11] Alexandre Gustavo Wermann, Marcelo Cardoso Bortolozzo, Eduardo Germano da Silva, Alberto Schaeffer-Filho, Luciano Paschoal Gaspary, and Marinho Barcellos. Astoria: A framework for attack simulation and evaluation in smart grids. In *NOMS 2016 - 2016 IEEE/IFIP Network Operations and Management Symposium*, pages 273–280, 2016.

[12] Joseph Yuen, Benjamin Turnbull, and Justin Hernandez. Visual analytics for cyber red teaming. In *2015 IEEE Symposium on Visualization for Cyber Security (VizSec)*, pages 1–8, 2015.