

OPEN THREAT RESEARCH

EMPOWERING THE INFOSEC COMMUNITY

Compartiendo Herramientas Open Source con el Mundo para el desarrollo de
Detecciones desde Casa

Quiénes Somos?



Amantes de la Colaboración y Fuente Abierta

Roberto Rodriguez 

@Cyb3rWard0g

- Microsoft Threat Intelligence Center (MSTIC)

Jose Rodriguez 

@Cyb3rPandaH

- MITRE - ATT&CK

- Colaboración Abierta ❤
- Threat Hunter Playbook
[@HunterPlaybook](#)
- Mordor [@Mordor_Project](#)
- OSSEM [@OSSEM_Project](#)
- Blacksmith & more..

Agenda

- 1) Qué Significa ser del lado Azul?
- 2) Una Metodología Basica
 - El Objetivo del R&D
 - Entendiendo al Adversario
 - Recolectando y Documentando Data
 - Analysis y Modelamiento de Datos
 - Validación y Documentación
 - Compartiendo con la Comunidad
- 3) Open Threat Research



Qué Significa ser del lado Azul?

Defensivo? Ofensivo?

El Lado Azul

- Constante adaptación y aprendizaje
- Entender al adversario y su comportamiento por medio de horas y horas de leer documentación y simulaciones
- Documentar y modelar data en relación a acciones ejecutadas por el adversario
- Planear estrategias de colección de data para tener mejor visibilidad
- Enteder el como mitigar amenazas en paralelo con el desarrollo de detecciones
- Entender el estado normal de una organización
- Responder a incidentes e identificar el alcance de el impacto

El Lado Azul

- Constante adaptación y aprendizaje
- **Aceptar diversidad en la industria**
- Entender al adversario y su comportamiento por medio de horas y horas de leer documentación y simulaciones
- **Identificar oportunidades de colaboración y aprendizaje**
- Documentar y modelar data en relación a acciones ejecutadas por el adversario
- Planear estrategias de colección de data para tener mejor visibilidad
- **Adoptar conceptos abiertos para facilitar contribución y aprendizaje**
- Enteder el como mitigar amenazas en paralelo con el desarrollo de detecciones
- Responder a incidentes e identificar el alcance de el impacto
- **Compartir conocimientos para empoderar a otros en la comunidad**

Aceptar y Adoptar Diversidad ...

Análisis de
Inteligencia

Respuesta a
Incidentes

Ingeniería Inversa



Simulación de
Adversarios

Análisis de Datos

Análisis Forense

Identificar Oportunidades de Collaboración



Adoptando Proyectos de Código Abierto



Compartir Conocimientos con la Comunidad



Colaboración en Conjunto

" If you want to go **fast**, go alone.
If you want to go **far**, go together"

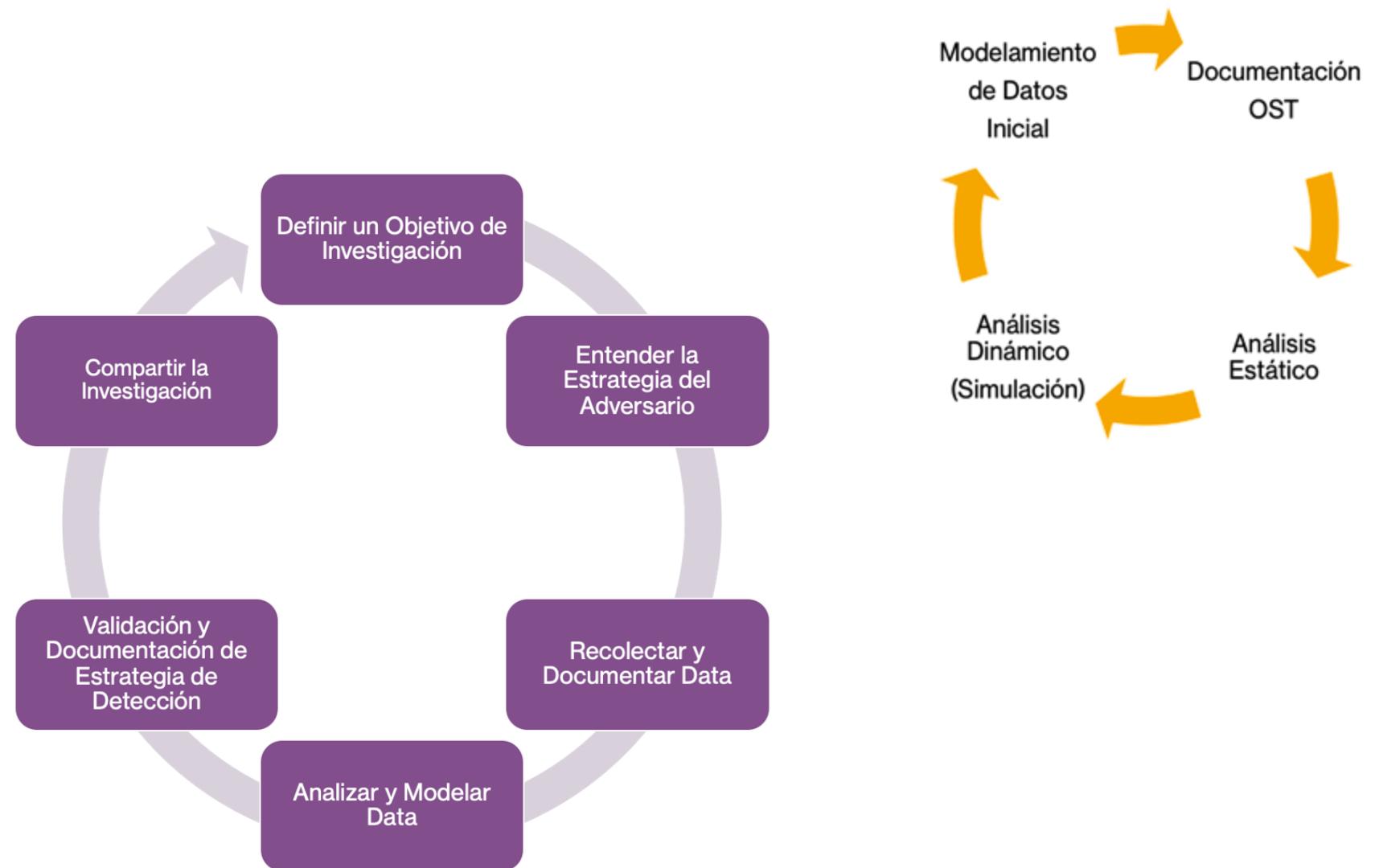
African Proverb

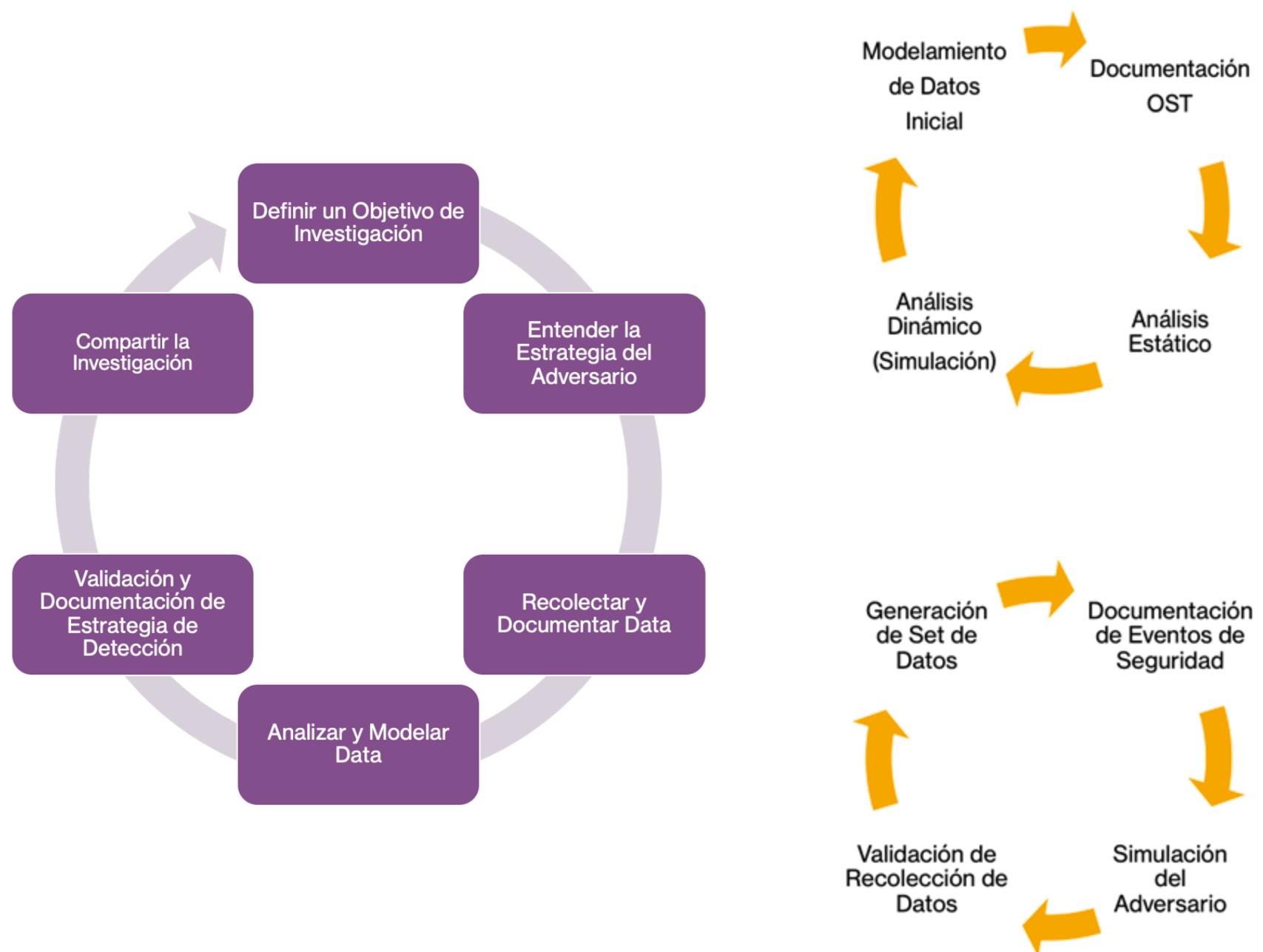
Una Metodología Basica

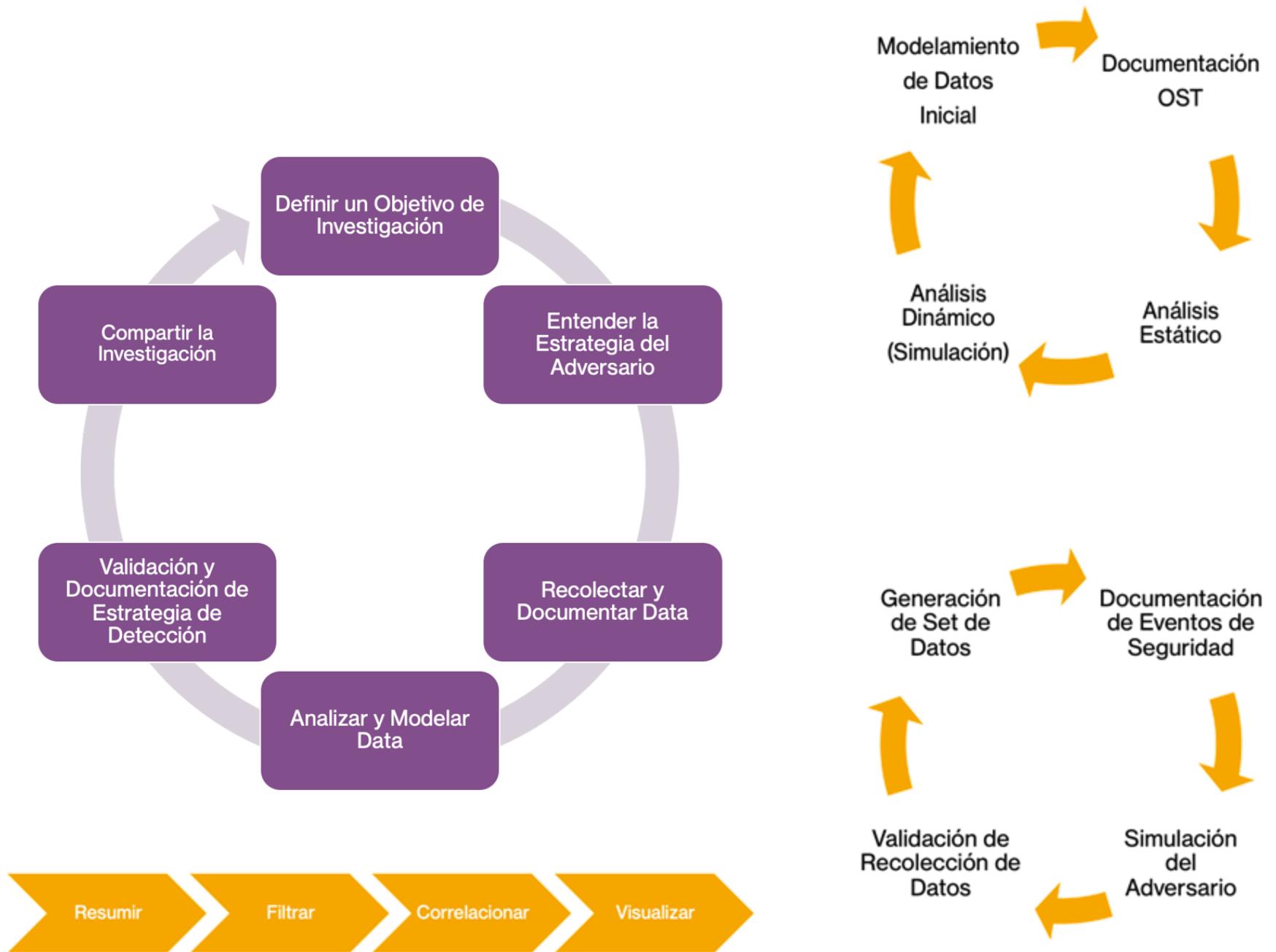
Colaborando, desarrollando
conceptos abiertos y compartiendo
durante la investigación y desarrollo
de detecciones!

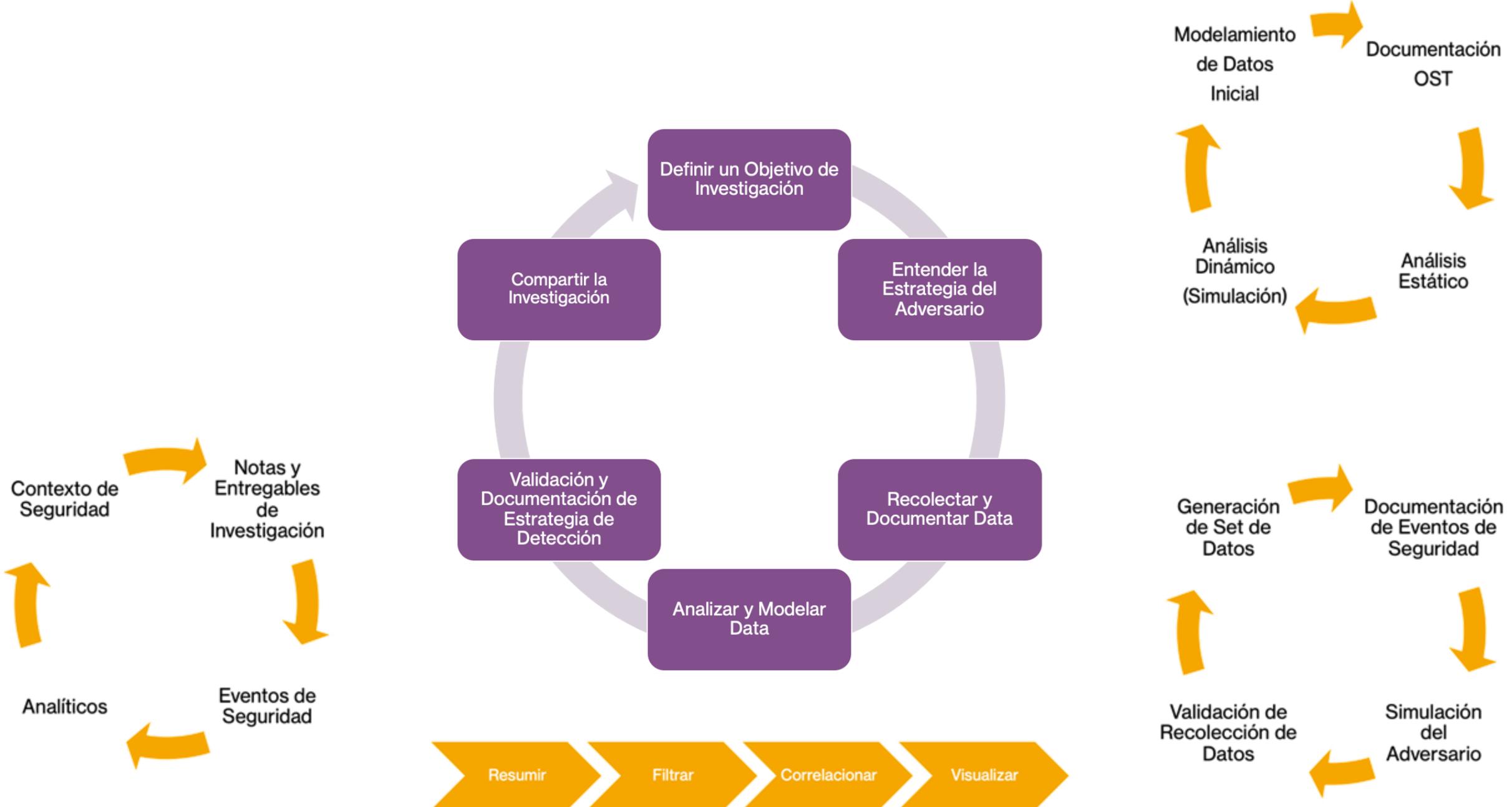


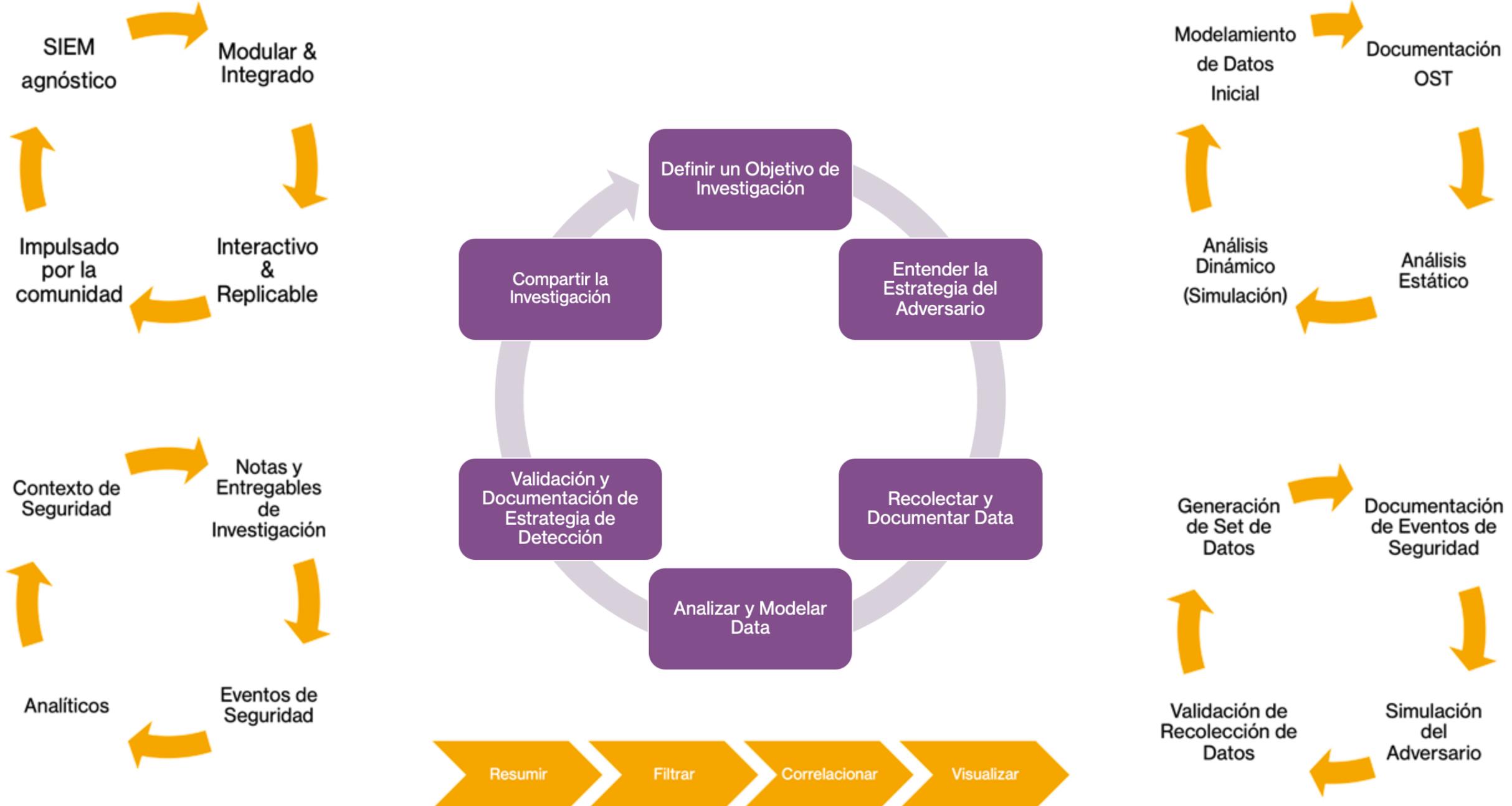


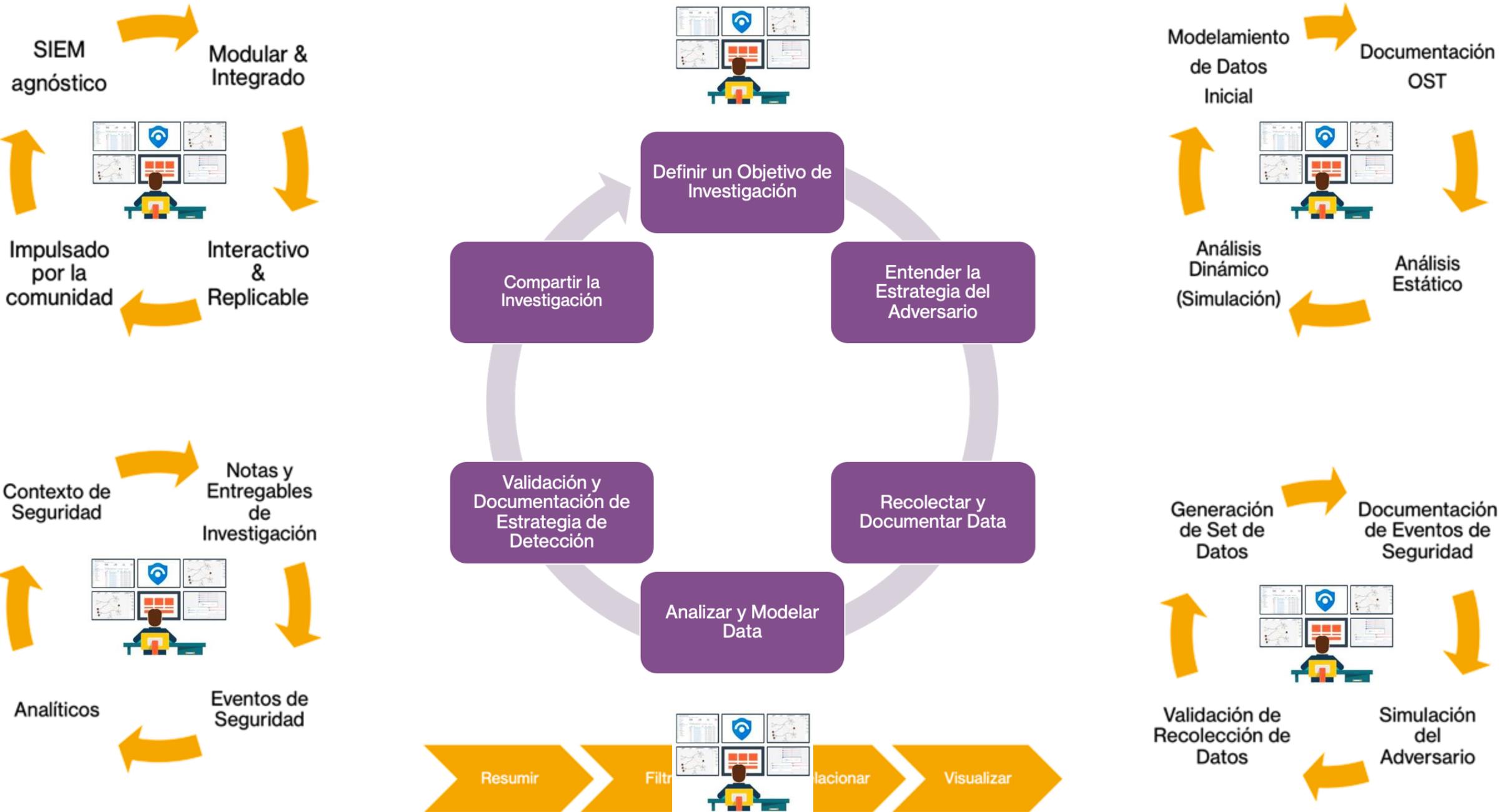
















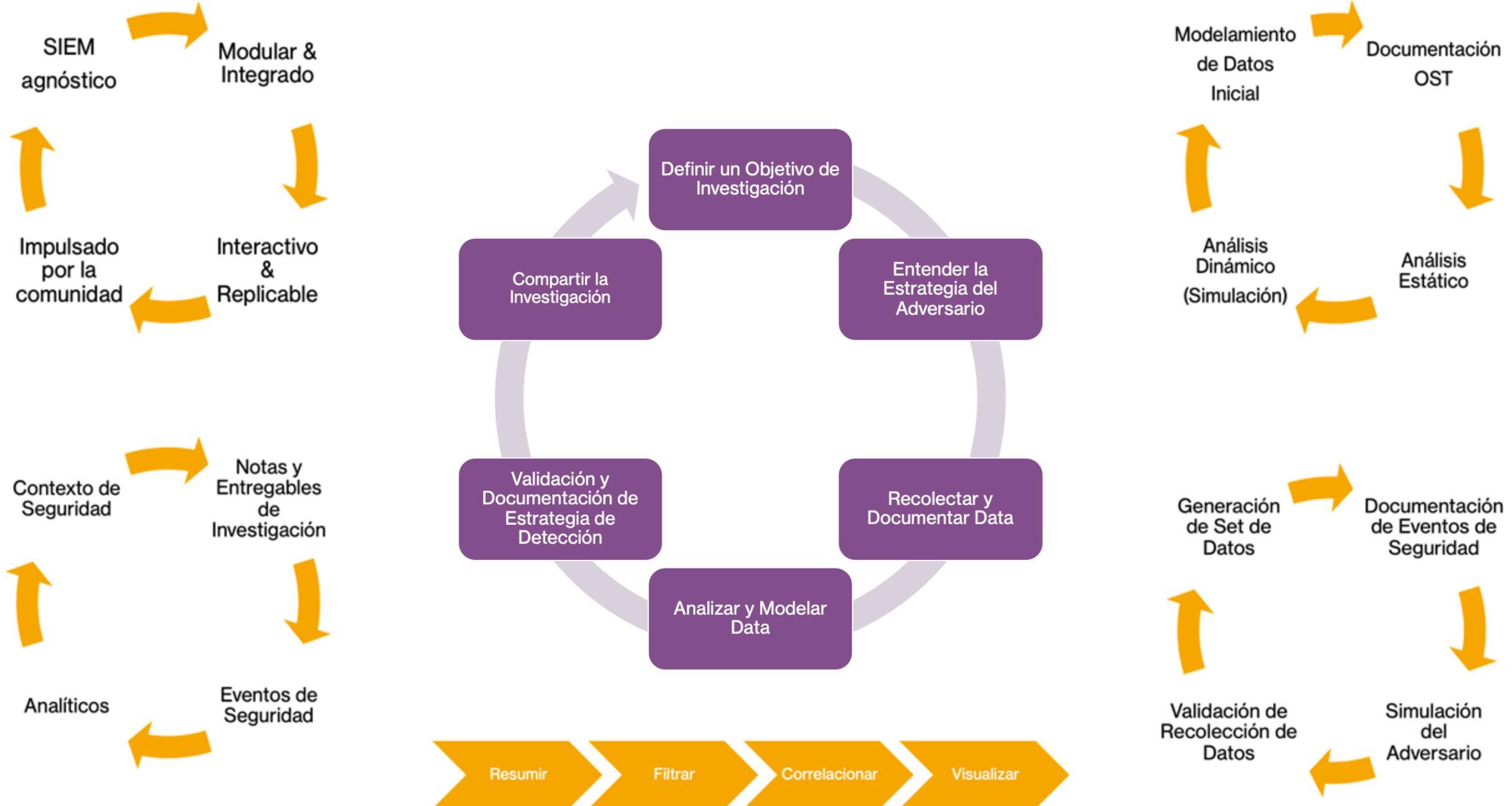


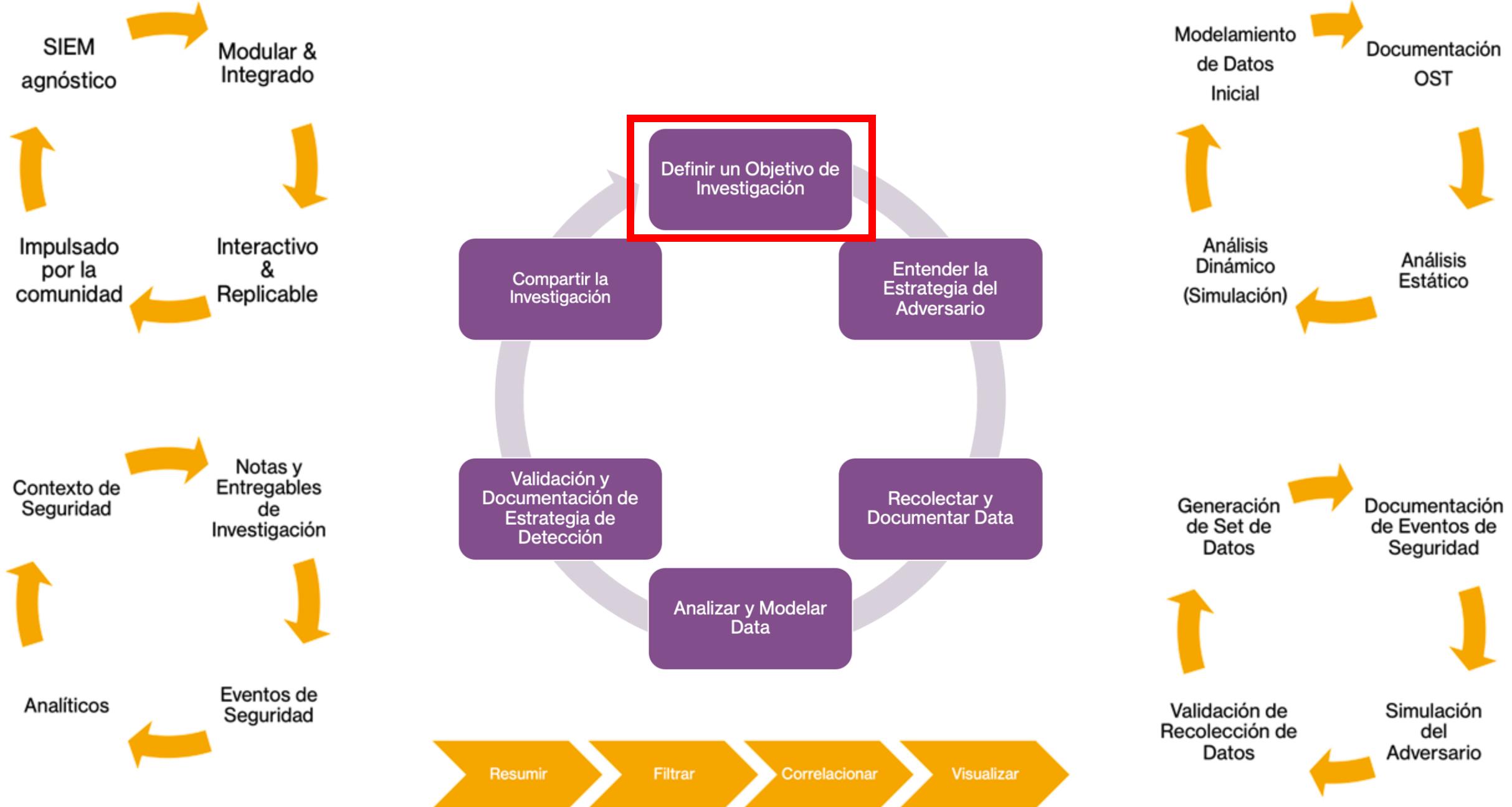


Reconoce tus
Fortalezas e
Identifica tu
Pasión

"Everybody is a Genius. But If You
Judge a Fish by Its Ability to Climb a
Tree, It Will Live Its Whole Life
Believing that It is Stupid"

Albert Einstein





ATT&CK Matrix for Enterprise

layouts ▾

show sub-techniques

hide sub-techniques

El Objetivo de la Investigación

Definiendo el alcance y tema
para R&D

Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection
Techniques	18 techniques	12 techniques	34 techniques	14 techniques	24 techniques	9 techniques	16 techniques
Code and Command Execution (7)	Account Manipulation (4)	Abuse Elevation Control Mechanism (4)	Abuse Elevation Control Mechanism (4)	Brute Force (4)	Account Discovery (4)	Exploitation of Remote Services	Archive Collected Data (3)
File and Application Execution for Persistence (2)	BITS Jobs	Access Token Manipulation (5)	Access Token Manipulation (5)	Credentials from Password Stores (3)	Application Window Discovery	Internal Spearphishing	Audio Capture
Process Execution (2)	Boot or Logon Autostart Execution (11)	BITS Jobs	Deobfuscate/Decode Files or Information	Exploitation for Credential Access	Browser Bookmark Discovery	Lateral Tool Transfer	Automated Collection
Script Execution (2)	Boot or Logon Initialization Scripts (5)	Boot or Logon Autostart Execution (11)	Direct Volume Access	Forced Authentication	Cloud Service Dashboard	Clipboard Data	Data from Cloud Storage Object
Tool and Agent Execution (2)	Browser Extensions	Boot or Logon Initialization Scripts (5)	Execution Guardrails (1)	Input Capture (4)	Cloud Service Discovery	Remote Service Session Hijacking (2)	Data from Information Repositories
Client Tools (2)	Compromise Client Software Binary	Create or Modify System Process (4)	Exploitation for Defense Evasion	Man-in-the-Middle (1)	Domain Trust Discovery	Remote Services (6)	Data from Local System
Agent Tools (2)	Create Account (3)	Event Triggered Execution (15)	File and Directory Permissions Modification (2)	File and Directory Permissions Modification (2)	File and Directory Discovery	Replication Through Removable Media	Data from Network Shared Drive
Network Tools (2)	Create or Modify System Process (4)	Exploitation for Privilege Escalation	Group Policy Modification	Group Policy Modification	Network Service Scanning	Software Deployment Tools	Data from Removable Media
File Tools (2)	Event Triggered Execution (15)	Group Policy Modification	Hide Artifacts (6)	Network Sniffing	Network Share Discovery	Taint Shared Content	Data Staged
Memory Tools (2)	External Remote Services	Hijack Execution Flow (11)	Hijack Execution Flow (11)	OS Credential Dumping (8)	Network Sniffing	Password Policy Discovery	Email Collection (3)
Network Tools (2)	Hijack Execution Flow (11)	Impair Defenses (6)	Impair Defenses (6)	Steal Application Access Token	Peripheral Device Discovery	Use Alternate Authentication Material (4)	Input Capture (4)
File Tools (2)	Hijack Execution Flow (11)	Indicator Removal on Host (6)	Indicator Removal on Host (6)	Steal or Forge Kerberos Tickets (3)	Permission Groups Discovery (3)		
Memory Tools (2)	Hijack Execution Flow (11)	Indirect Command	Indirect Command				

ATT&CK® framework

ATT&CK Matrix for Enterprise

layouts ▾
show sub-techniques
hide sub-techniques

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Command and Control	Exfiltration	Impact
9 techniques	10 techniques	18 techniques	12 techniques	34 techniques	14 techniques	24 techniques	9 techniques	16 techniques	16 techniques	9 techniques	13 techniques
Drive-by Compromise	Command and Scripting Interpreter (7)	Account Manipulation (4)	Abuse Elevation Control Mechanism (4)	Brute Force (4)	Account Discovery (4)	Exploitation of Remote Services	Archive Collected Data (3)	Application Layer Protocol (4)	Automated Exfiltration	Account Access Removal	
Exploit Public-Facing Application	Exploitation for Client Execution	BITS Jobs	Access Token Manipulation (5)	Credentials from Password Stores (3)	Application Window Discovery	Internal Spearphishing	Audio Capture	Communication Through Removable Media	Data Transfer Size Limits	Data Destruction	
External Remote Services	Inter-Process Communication (2)	Boot or Logon Autostart Execution (11)	BITS Jobs	Exploitation for Credential Access	Browser Bookmark Discovery	Lateral Tool Transfer	Automated Collection	Clipboard Data	Exfiltration Over Alternative Protocol (3)	Data Encrypted for Impact	
Hardware Additions	Native API	Boot or Logon Initialization Scripts (5)	Deobfuscate/Decode Files or Information	Forced Authentication	Cloud Service Dashboard	Remote Service Session Hijacking (2)	Data from Cloud Storage Object	Dynamic Resolution (3)	Exfiltration Over C2 Channel	Data Manipulation (3)	
Phishing (3)	Scheduled Task/Job (5)	Browser Extensions	Direct Volume Access	Execution Guardrails (1)	Cloud Service Discovery	Domain Trust Discovery	Data from Information Repositories (2)	Encrypted Channel (2)	Exfiltration Over Other Network Medium (1)	Defacement (2)	
Replication Through Removable Media	Shared Modules	Compromise Client Software Binary	Exploitation for Defense Evasion	Input Capture (4)	File and Directory Discovery	File and Directory Discovery	Data from Local System	Fallback Channels	Inhibit System Recovery	Disk Wipe (2)	
Supply Chain Compromise (3)	Software Deployment Tools	Create Account (3)	Event Triggered Execution (15)	Man-in-the-Middle (1)	Network Service Scanning	Network Share Discovery	Data from Network Shared Drive	Ingress Tool Transfer	Network Denial of Service (2)	Firmware Corruption	
Trusted Relationship	System Services (2)	Create or Modify System Process (4)	Exploitation for Privilege Escalation	Modify Authentication Process (3)	Network Sniffing	Network Sniffing	Data from Removable Media	Multi-Stage Channels	Resource Hijacking		
Valid Accounts (4)	User Execution (2)	Event Triggered Execution (15)	Group Policy Modification	OS Credential Dumping (8)	Passport Policy Discovery	Taint Shared Content	Non-Application Layer Protocol	Non-Standard Port Protocol Tunneling	Service Stop		
	Windows Management Instrumentation	External Remote Services	Hijack Execution Flow (11)	Steal Application Access Token	Peripheral Device Discovery	Use Alternate Authentication Material (4)	Email Collection (3)	Proxy (4)	Transfer Data to Cloud Account	System Shutdown/Reboot	
		Hijack Execution Flow (11)	Process Injection (11)	Steal or Forge Kerberos Tickets (3)	Permission Groups Discovery (3)	Input Capture (4)	Man in the Browser	Remote Access Software			
		Implant Container Image	Scheduled Task/Job (5)	Indirect Command Execution	Process Discovery	Query Registry	Man-in-the-Middle (1)	Traffic Signaling (1)			
		Office Application Startup (6)	Valid Accounts (4)	Masquerading (6)	Query Registry	Remote System Discovery	Screen Capture	Web Service (3)			
		Pre-OS Boot (3)		Two-Factor Authentication Interception	Software Discovery (1)	System Information Discovery					
		Scheduled Task/Job (5)		Unsecured Credentials (6)							

ATT&CK® framework

Home > Techniques > Enterprise > Hijack Execution Flow > DLL Search Order Hijacking

Hijack Execution Flow: DLL Search Order Hijacking

Other sub-techniques of Hijack Execution Flow (11)

Adversaries may execute their own malicious payloads by hijacking the search order used to load DLLs. Windows systems use a common method to look for required DLLs to load into a program. [1] Hijacking DLL loads may be for the purpose of establishing persistence as well as elevating privileges and/or evading restrictions on file execution.

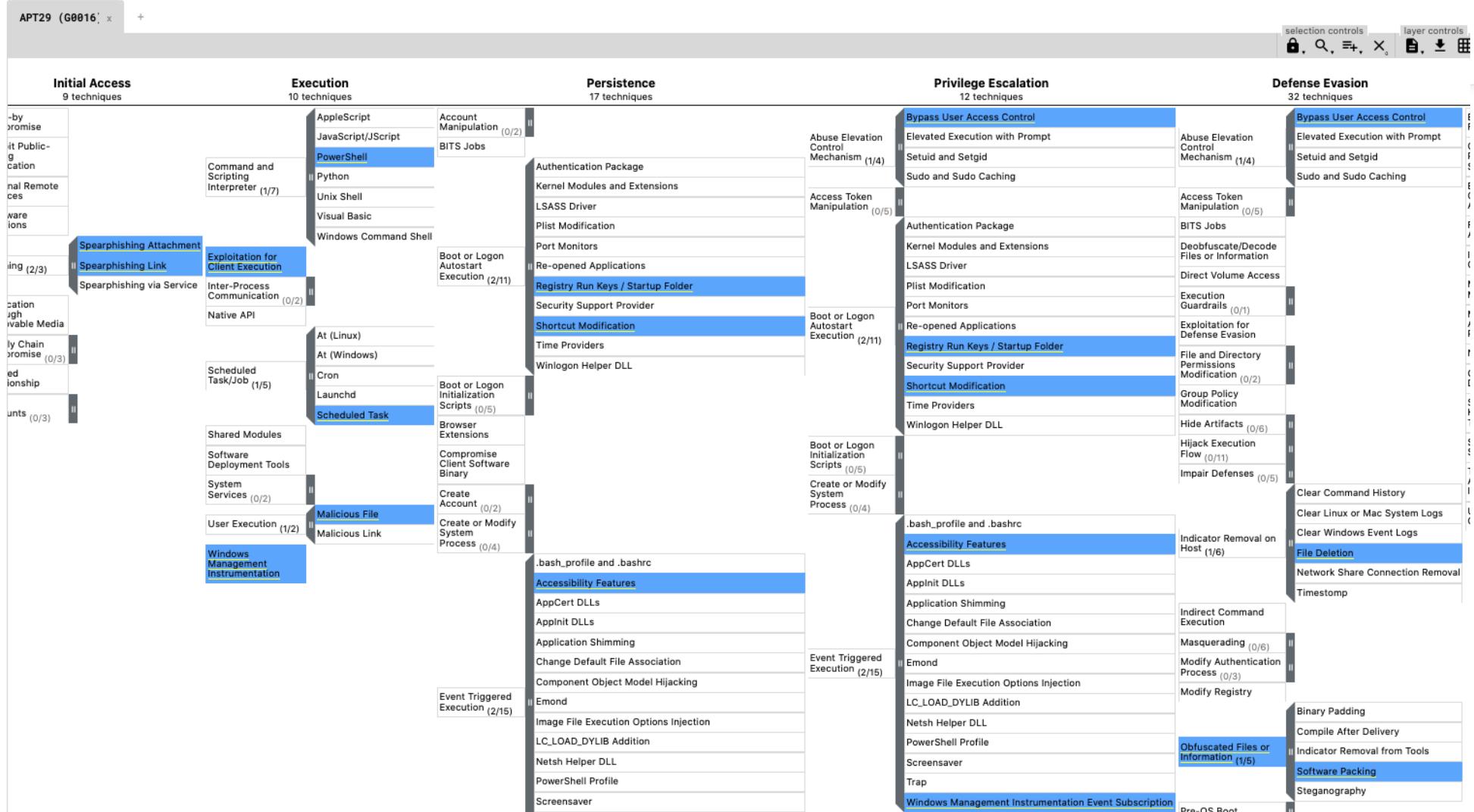
There are many ways an adversary can hijack DLL loads. Adversaries may plant trojan dynamic-link library files (DLLs) in a directory that will be searched before the location of a legitimate library that will be requested by a program, causing Windows to load their malicious library when it is called for by the victim program. Adversaries may also perform DLL preloading, also called binary planting attacks, [2] by placing a malicious DLL with the same name as an ambiguously specified DLL in a location that Windows searches before the legitimate DLL. Often this location is the current working directory of the program. Remote DLL preloading attacks occur when a program sets its current directory to a remote location such as a Web share before loading a DLL. [3]

Adversaries may also directly modify the way a program loads DLLs by replacing an existing DLL or modifying a .manifest or .local redirection file, directory, or junction to cause the program to load a different DLL. [4] [5] [6]

ID: T1574.001
Sub-technique of: [T1574](#)
Tactics: Persistence, Privilege Escalation, Defense Evasion
Platforms: Windows
Data Sources: DLL monitoring, File monitoring, Process command-line parameters, Process monitoring
CAPEC ID: [CAPEC-471](#)
Contributors: Stefan Kanthak; Travis Smith, Tripwire
Version: 1.0
Created: 13 March 2020
Last Modified: 26 March 2020

[Version Permalink](#)

Evaluaciones via ATT&CK®



<https://mitre-attack.github.io/attack-navigator/enterprise/#layerURL=https%3A%2F%2Fattack.mitre.org%2Fgroups%2FG0016%2FG0016-enterprise-layer.json>

Ingeligencia desde la comunidad

MDSec
@MDSecLabs

Part 1 in the "I Like to Move It" series on lateral movement by @domchell is now live... mdsec.co.uk/2020/09/i-like... #redteam



9:26 AM · Sep 1, 2020 · Twitter Web App

120 Retweets 4 Quote Tweets 6 Replies 103 Likes

MDSec @MDSecLabs · Sep 1, 2020

Part 2 in @domchell's series on lateral movement. Check out the blog mdsec.co.uk/2020/09/i-like...

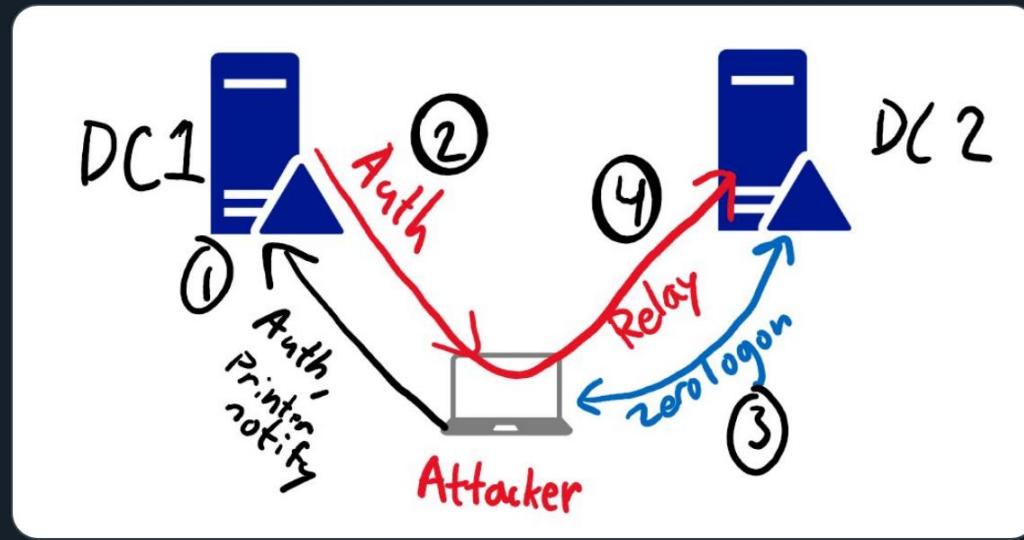


9:26 AM · Sep 1, 2020 · Twitter Web App

120 Retweets 4 Quote Tweets 6 Replies 103 Likes

Dirk-jan
 @_dirkjan

New blog: A different way of abusing Zerologon. No more password reset needed: using the printer bug with Zerologon to relay to DRSUAPI and DCSync directly with ntlmrelayx: dirkjanm.io/a-different-wa...
Code: github.com/dirkjanm/CVE-2...



3:33 PM · Sep 24, 2020 · Twitter Web App

100 Likes

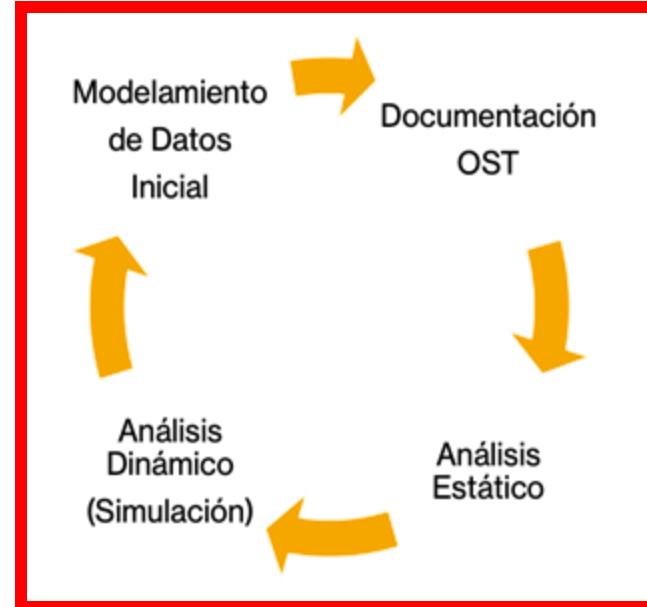
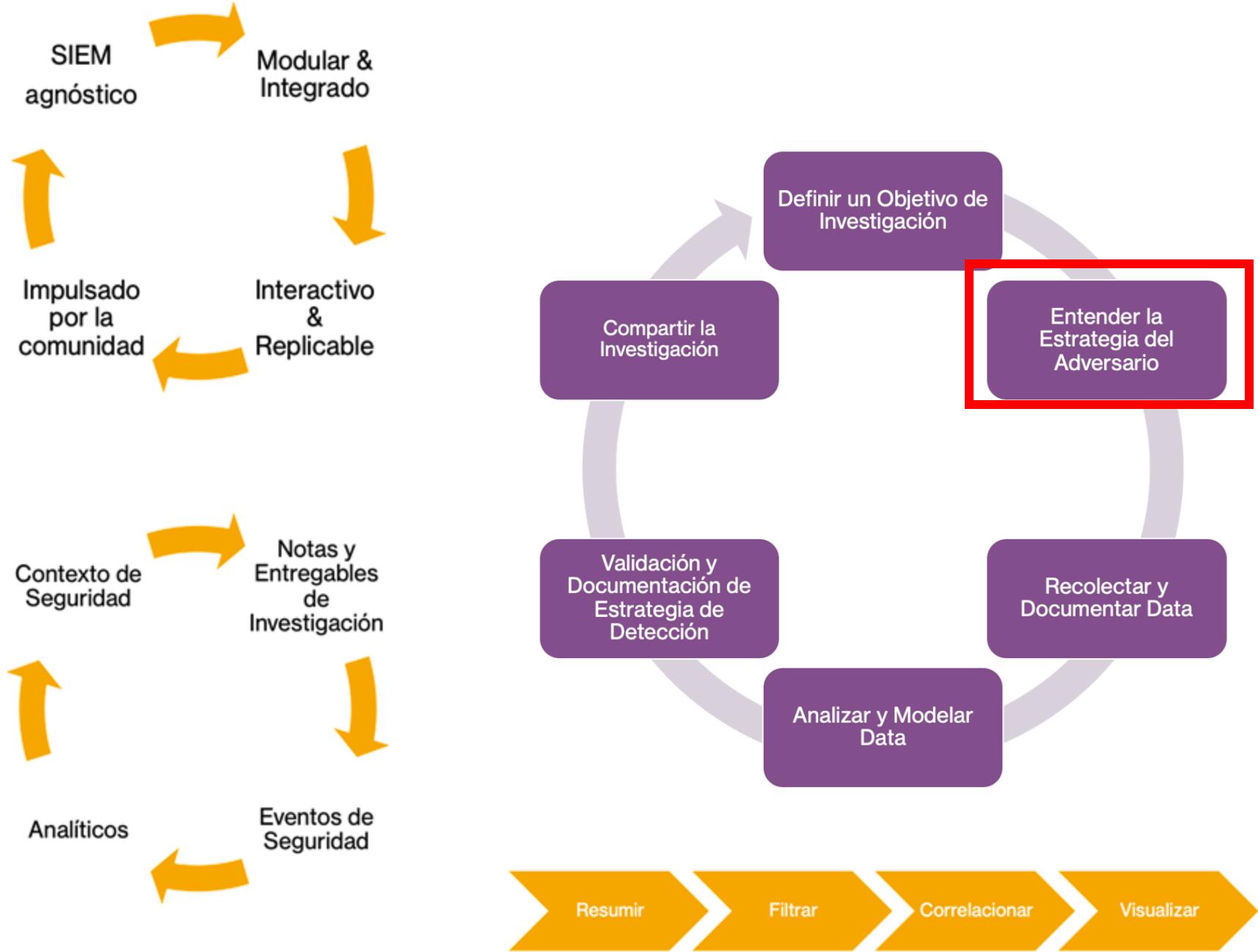
Benjamin Delpv

se with #zerologon / CVE-2020-1472 : support and a lots of love inside ❤️

I (fast and supports unauthenticated on m...)

Web App

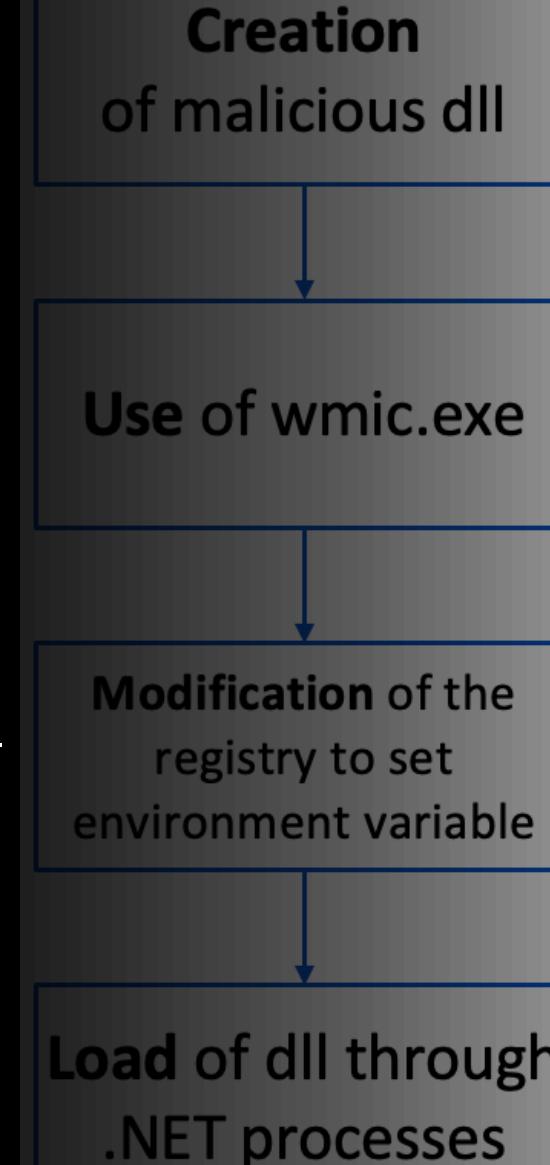
1K Likes



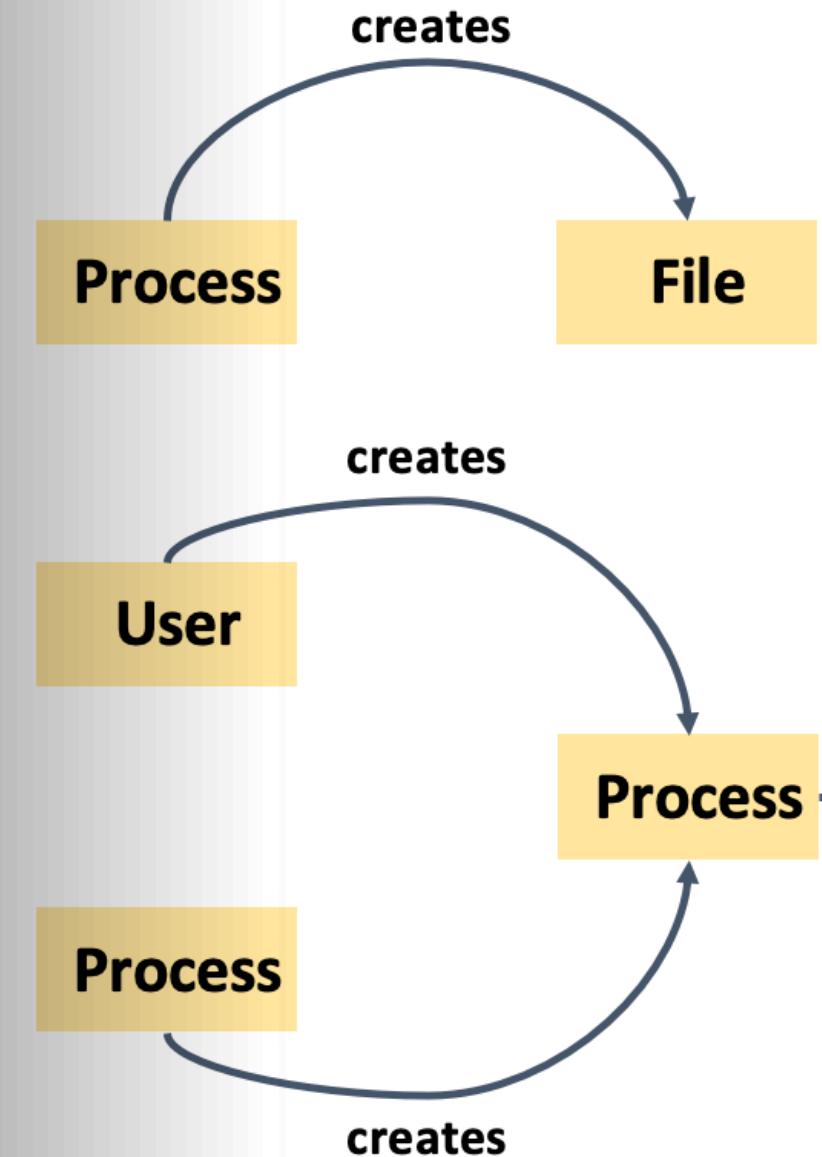
Entendiendo al Adversario

La estrategia y el
comportamiento

Adversary



Model



Movimiento Lateral con DCOM via ExecuteExcel4Macro



Aprendiendo el Comportamiento

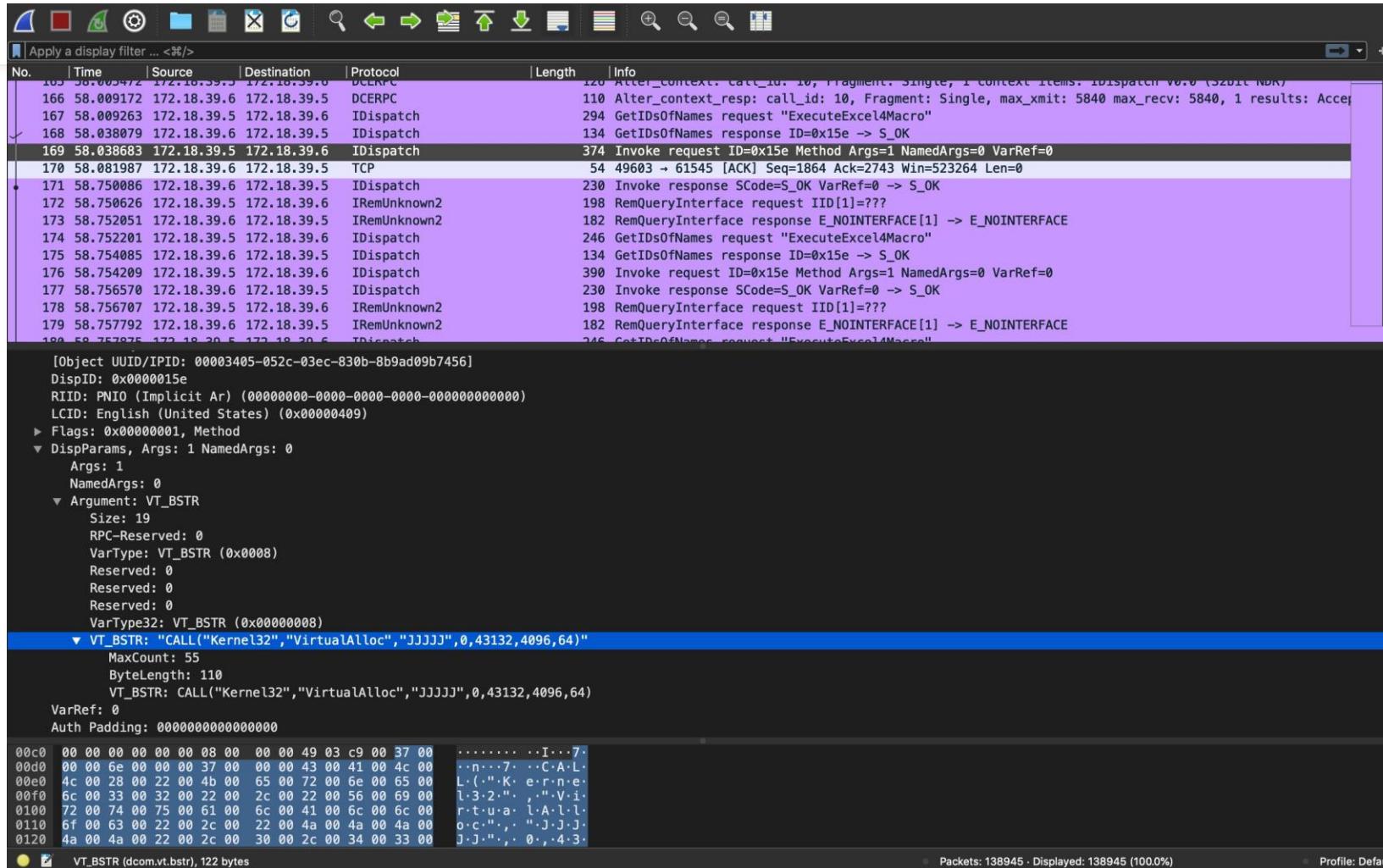
In order to weaponise this technique, we ideally want it to execute in a fileless manner. As explained by Outflank, XLM code has direct access to the Win32 API so we can leverage this to execute shellcode by writing it to memory and starting a new thread:

```
var memaddr = Convert.ToDouble(excel.GetType().InvokeMember("ExecuteExcel4Macro", BindingFlags.InvokeMethod, null, excel, new object[] { }));
var startaddr = memaddr;

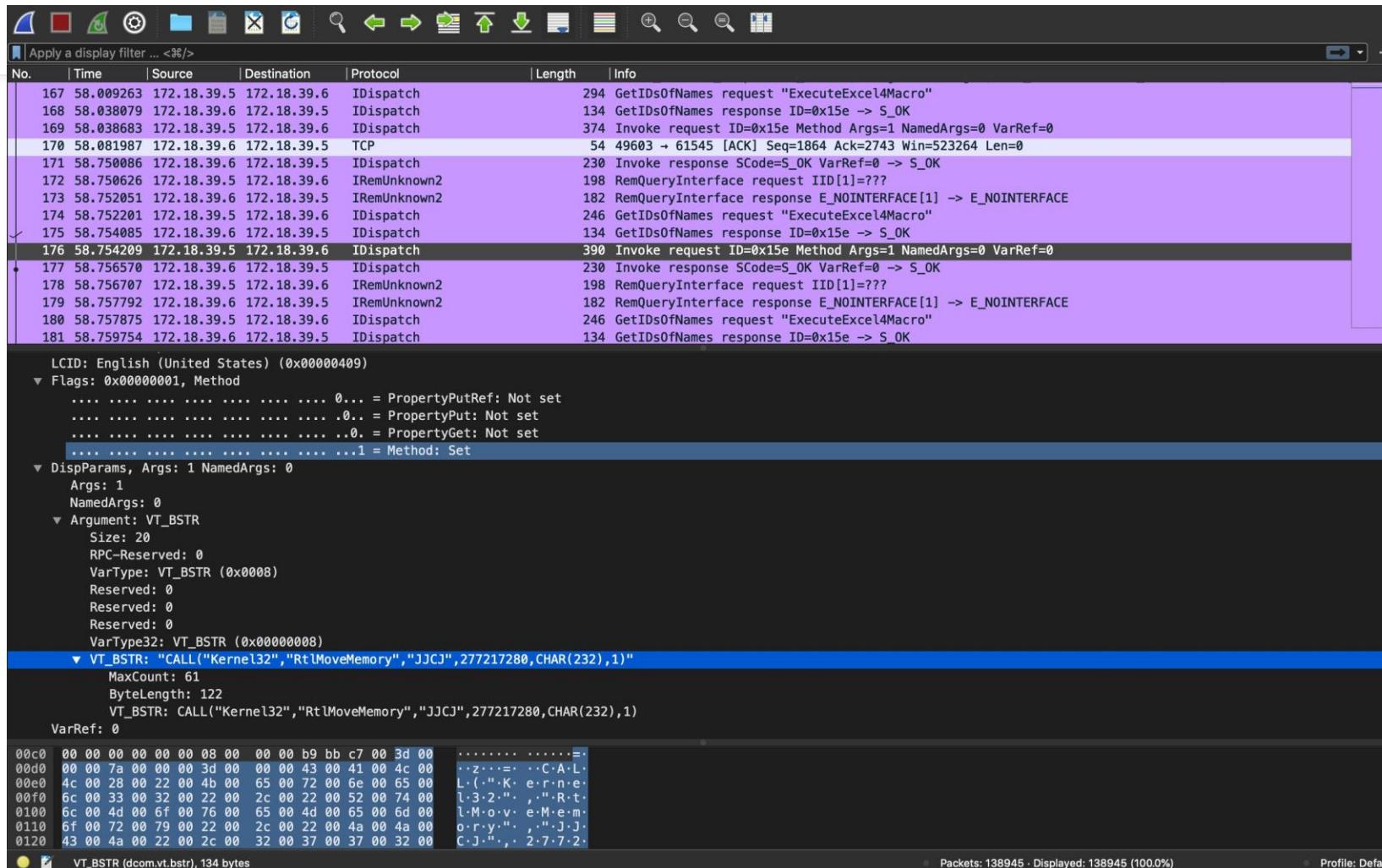
foreach (var b in shellcode) {
    var cb = String.Format("CHAR({0})", b);
    var macrocode = "CALL(\\"Kernel32\\",\\"RtlMoveMemory\\",\\"JJCJ\\", " + memaddr + "," + cb + ",1";
    excel.GetType().InvokeMember("ExecuteExcel4Macro", BindingFlags.InvokeMethod, null, excel, new object[] { macrocode });
    memaddr++;
}
excel.GetType().InvokeMember("ExecuteExcel4Macro", BindingFlags.InvokeMethod, null, excel, new object[] {});
```

This of course can be improved to do remote process injection or speed up execution by moving the bytes in chunks.

Análisis básico dinámico



Análisis básico dinámico



Entendiendo la estrategia del adversario



<https://github.com/OTRF/Blacksmith>



<https://github.com/OTRF/SimuLand>



<https://github.com/OTRF/mordor>

BlackSmith



- Este proyecto se enfoca en proveer plantillas dinámicas y de fácil uso para investigadores de ciberseguridad con el objetivo de modelar y proveer recursos para la implementacion automatica de aplicaciones y ambientes de red en la nube.
- AWS CloudFormation & Microsoft Azure Resource Manager
- <https://github.com/OTRF/Blacksmith>

BlackSmith



master ➔ Blacksmith / templates / azure / Win10-DC / nestedtemplates / customScriptExtension.json

Cyb3rWard0g Sentinel2Go Draft Win10-DC ...

1 contributor

46 lines (46 sloc) | 1.41 KB

```
1 {
2   "$schema": "https://schema.management.azure.com/schemas/2015-01-01/deploymentTemplate.json#",
3   "contentVersion": "1.0.0.0",
4   "parameters": {
5     "vmName": {
6       "type": "string",
7       "minLength": 1,
8       "metadata": {
9         "description": "Name of the windows machine to run scripts on"
10      }
11    },
12    "extensionName": {
13      "type": "string"
14    },
15    "fileUris": {
16      "type": "array"
17    },
18    "commandToExecute": {
19      "type": "string"
20    },
21    "location": {
22      "type": "string",
```

master ➔ Blacksmith / templates / azure / Win10-DC / scripts / Set-AD.ps1

Cyb3rWard0g Updated Docs & Master compatible

1 contributor

15 lines (12 sloc) | 368 Bytes

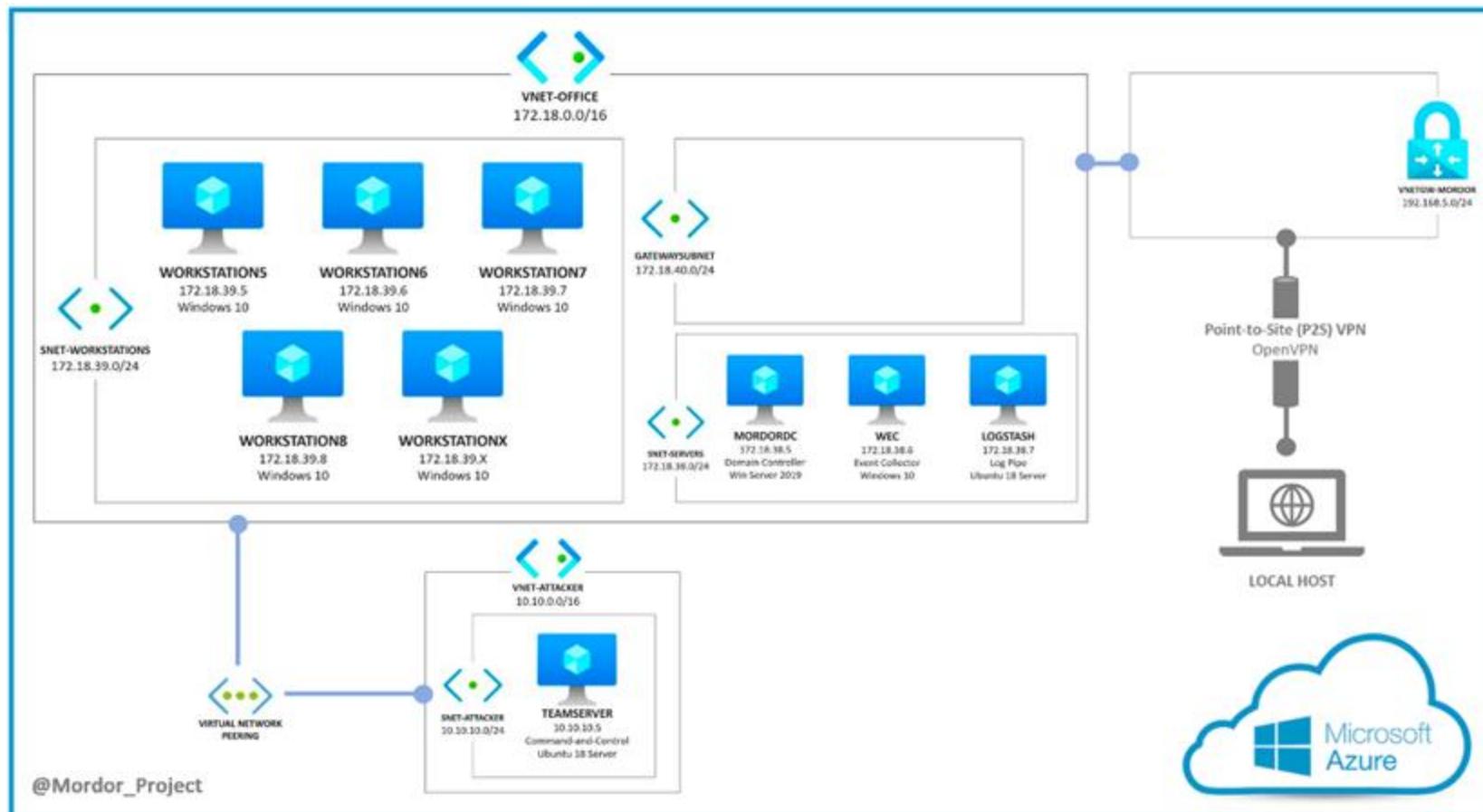
```
1 # Author: Roberto Rodriguez (@Cyb3rWard0g)
2 # License: GPL-3.0
3
4 [CmdletBinding()]
5 param(
6   [Parameter(Mandatory=$true, Position=1)]
7   [string]$domainFQDN,
8
9   [Parameter(Mandatory=$true, Position=2)]
10  [string]$dcVMName
11 )
12
13 & .\Set-0Us.ps1 -domainFQDN $domainFQDN
14 & .\Add-DomainUsers.ps1 -domainFQDN $domainFQDN -dcVMName $dcVMName
15 & .\Set-AuditSAMRemoteCalls.ps1
```

SimuLand

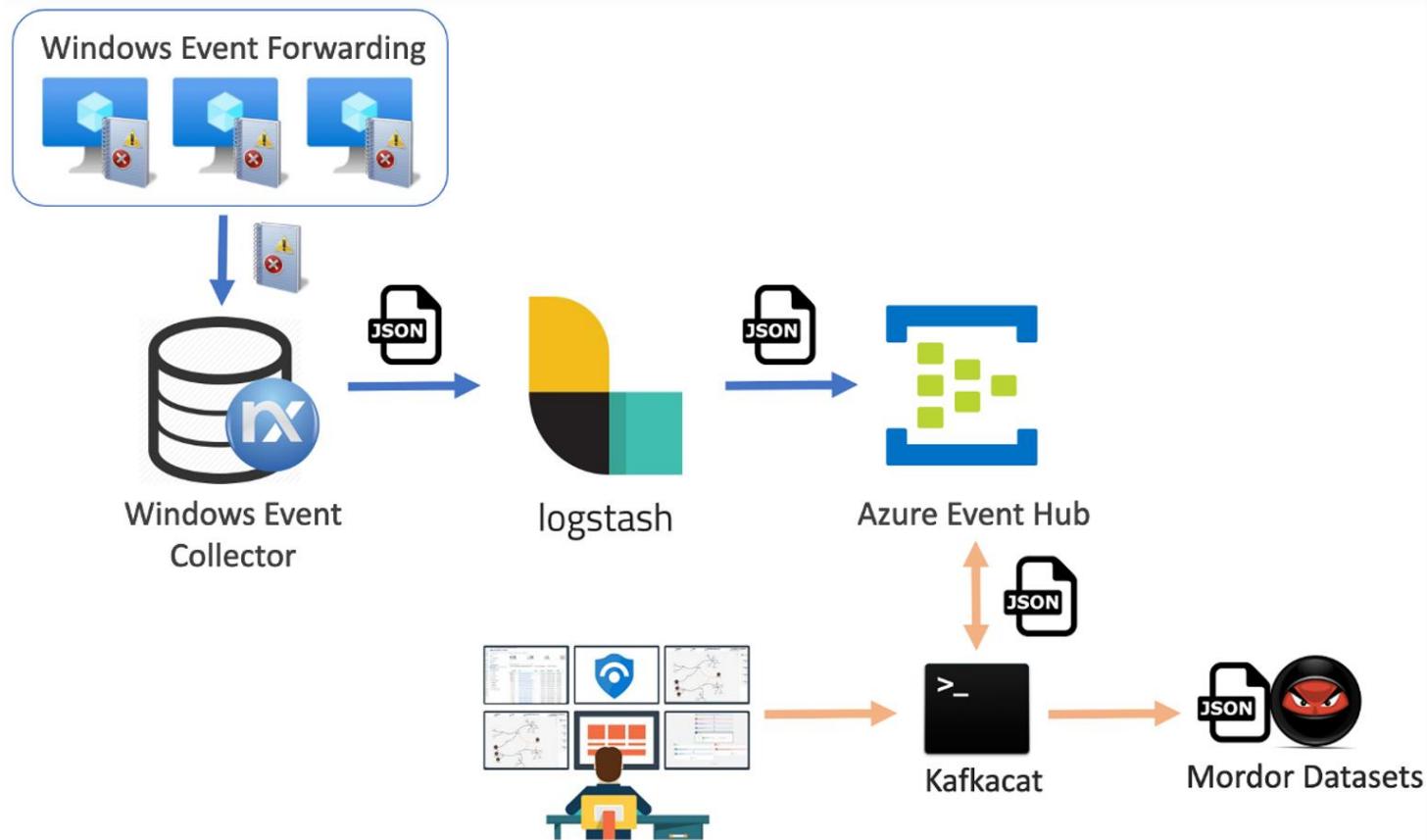


- **Cloud templates y scripts:**
 - Simulación de adversarios
 - Generación-recolección de datos
 - Enfocado en el aprendizaje de la estrategia del adversario desde una perspectiva de datos.
- **Múltiples ambientes modulares** que permiten la adaptación de requerimientos específicos de un tema específico (research).
- <https://github.com/OTRF/SimuLand>

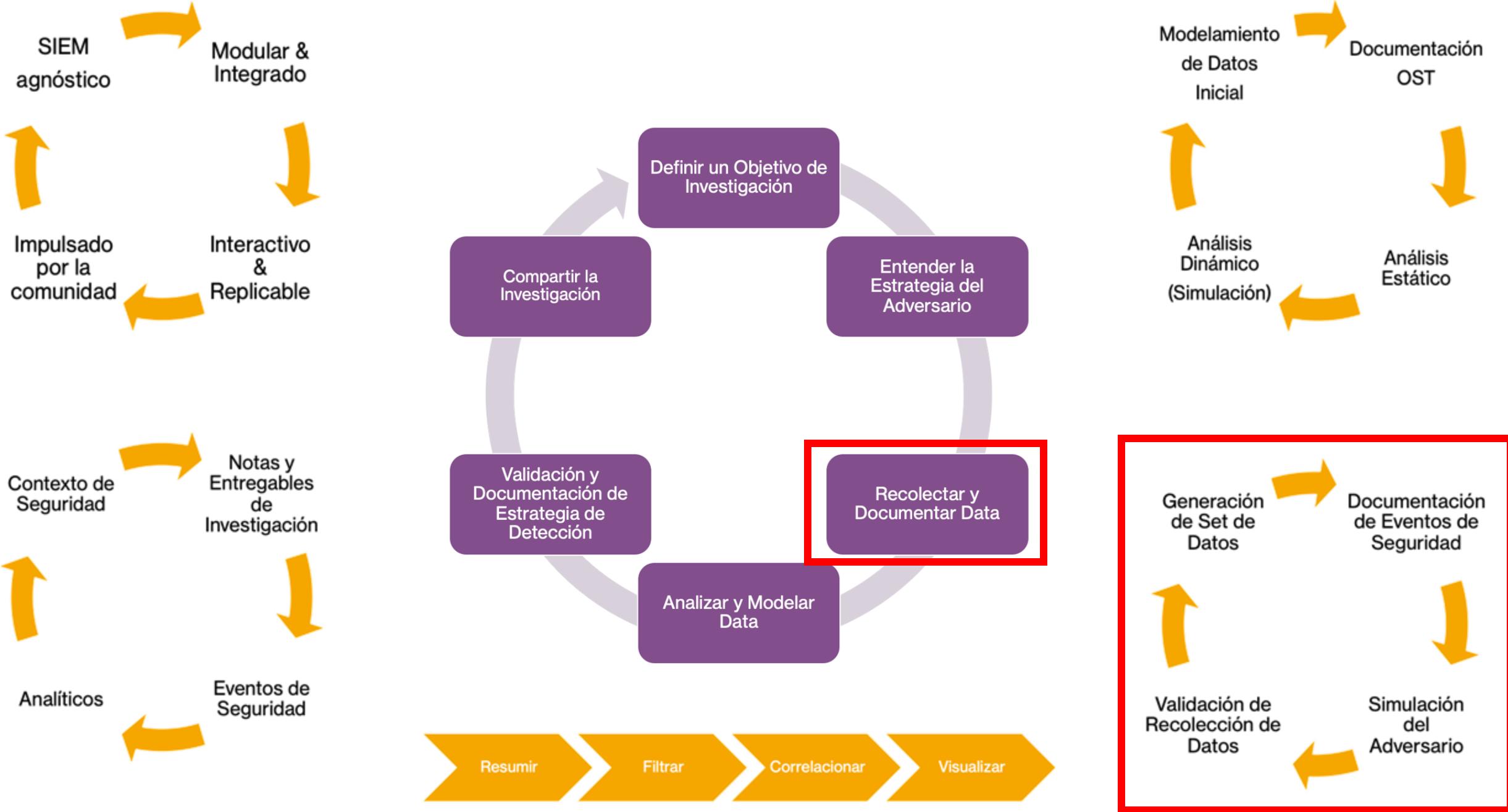
The Shire (Windows)



The Shire (Windows)



Simula -> Exporta



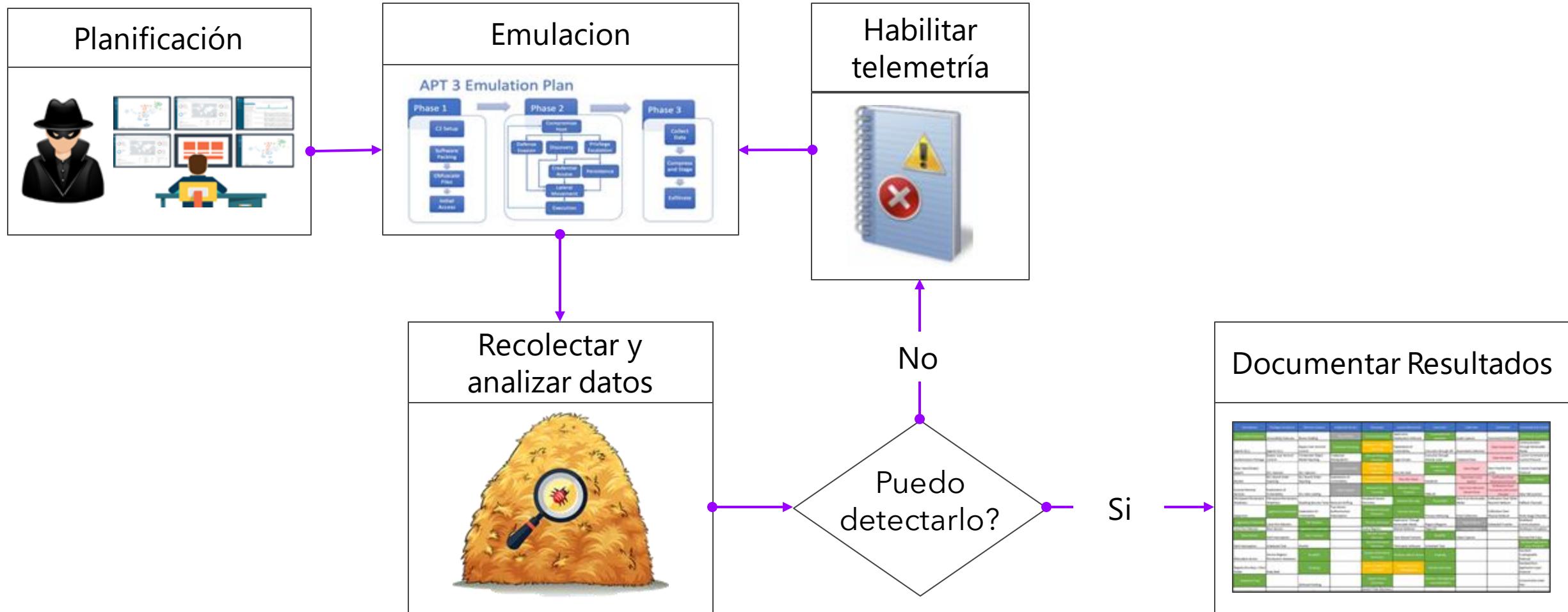


Recolectando y documentando data

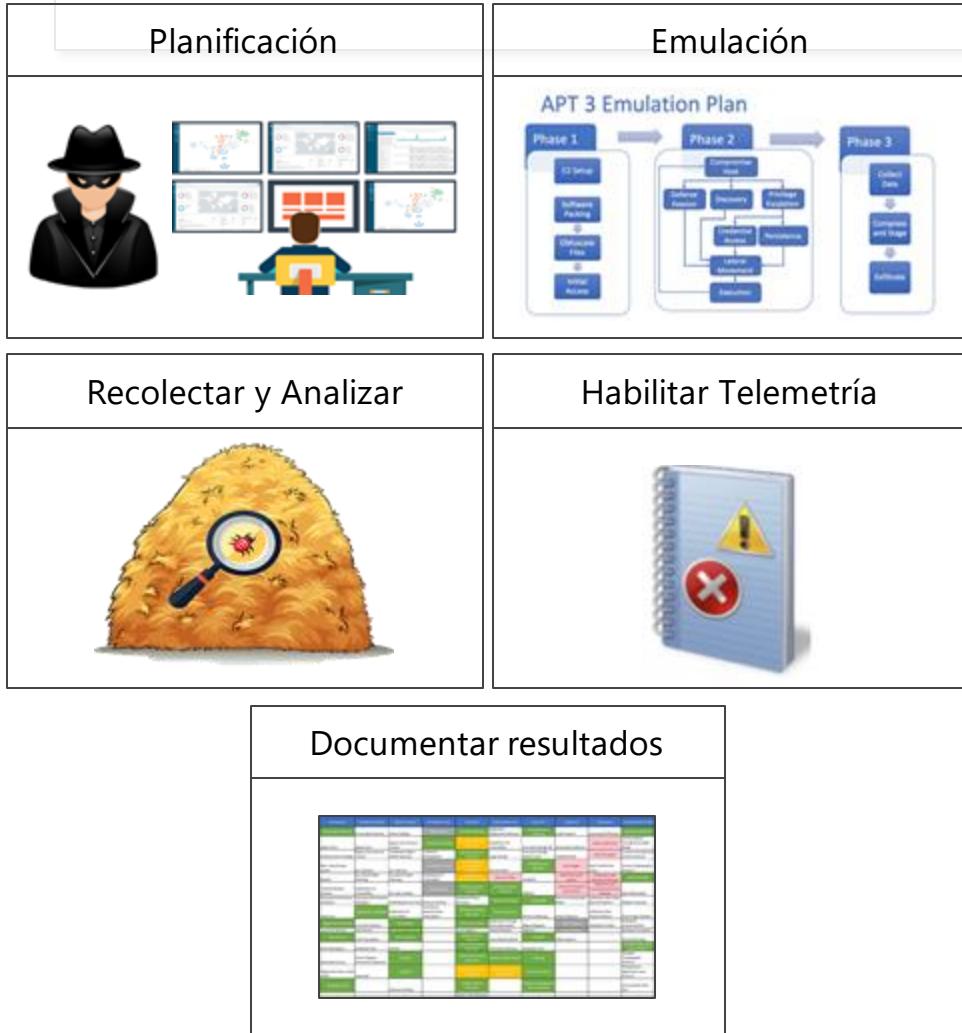
Generando data en Casa



Actividades Básicas al simular al adversario



Cómo? Dónde? Cuándo?



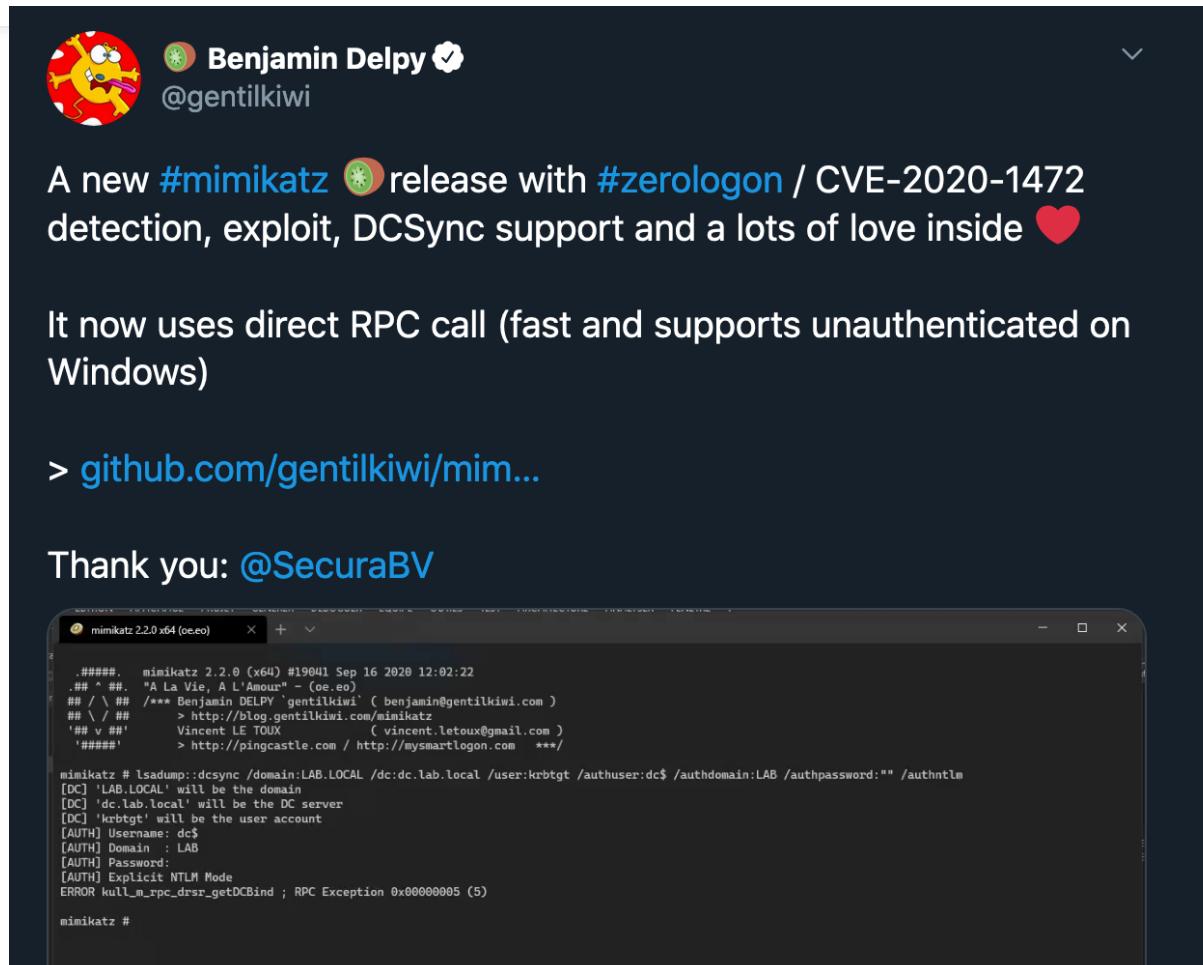
Desafíos

Puede requerir alta inversión de **tiempo** y **tecnología**

Pueda que tengas que escribir tus propias herramientas

Configurar la colección de datos es más que un solo comando o switch

Ejemplos? Por que es complicado?



Ejemplos? Por que es complicado?



Benjamin Delpy ✅
@gentilkiwi

A new **#mimikatz** 🍀 release with **#zerologon** detection, exploit, DCSync support and a lo...

It now uses direct RPC call (fast and supports Windows)

> github.com/gentilkiwi/mim...

Thank you: [@SecuraBV](#)

```
mimikatz 2.2.0 x64 (oe.eo) x + ^

#####
# ## mimikatz 2.2.0 (x64) #19041 Sep 16 2020 12:02:22
# ## '## "A La Vie, A L'Amour" - (oe.eo)
# ## '## /*** Benjamin DELPY gentilkiwi' ( benjamin@gentilkiwi.com )
# ## '## > http://blog.gentilkiwi.com/mimikatz
# ## '## '## Vincent Le Toux ( vincent.letoux@gmail.com )
# ## '## > http://pingcastle.com / http://nsmartlogon.com ***/
#####

mimikatz # lsadump::dcsync /domain:LAB.LOCAL /dc:dc.lab.local /user:krbtgt /authuser:dc$ /authdomain:L
[DC] 'LAB.LOCAL' will be the domain
[DC] 'dc.lab.local' will be the DC server
[DC] 'krbtgt' will be the user account
[AUTH] Username: dc$ 
[AUTH] Domain : LAB
[AUTH] Password: 
[AUTH] Explicit NTLM Mode
ERROR_krll_m_rpc_drsr_getDCBind ; RPC Exception 0x00000005 (5)

mimikatz #
```

Windows Server products & resources

(-) Windows Server 2019

Evaluations | **180 days**

In addition to your trial experience of Windows Server 2019, you can download a new feature on demand for Server Core, the App Compatibility FOD. This FOD contains additional features from the Desktop Experience to improve the compatibility of Server Core for apps and tools used for troubleshooting and debugging. Windows features on demand can be added to images prior to deployment or to actively running computers, using the DISM command. Learn more about the [Server Core App Compatibility FOD](#). Download this [FOD](#). To learn more about FODs in general, and the DISM command, please visit [DISM Capabilities Package Servicing](#).

(-) [Start your evaluation](#)

Please select your evaluation experience:

Azure

ISO

VHD

Continue

Ejemplos? Por que es complicado?

A new **#mimikatz** 🍀 release with **#zerologon** / CVE-202 detection, exploit, DCSync support and a lots of love ins

It now uses direct RPC call (fast and supports unauthen Windows)

> github.com/gentilkiwi/mim...

Thank you: [@SecuraBV](#)

```
mimikatz 2.2.0 x64 (oe.eo)
#####
    mimikatz 2.2.0 (x64) #19041 Sep 16 2020 12:02:22
.##. "#. "A La Vie, A L'Amour" - (oe.eo)
##( DC ) ##. "/*** Benjamin DELPY gentilkiwi' ( benjamin@gentilkiwi.com )
##( DC ) ##. > http://blog.gentilkiwi.com/mimikatz
##( DC ) ##. "krbtgt" will be the user account
##( DC ) ##. "krbtgt" will be the DC server
[AUTH] Username: dc5
[AUTH] Domain : LAB
[AUTH] Password:
[AUTH] Explicit NTLM Mode
ERROR_krnl_m_rpc_drsr_getDCBind ; RPC Exception 0x00000005 (5)

mimikatz #
```

Windows Server SKU ⓘ

2019-Datacenter

Windows Server Version ⓘ

latest

Windows Server Vm Size ⓘ

17763.1217.2005081535

17763.1282.2006061952

C2Framework ⓘ

17763.1339.2007101755

Ubuntu SKU ⓘ

17763.1397.2008070242

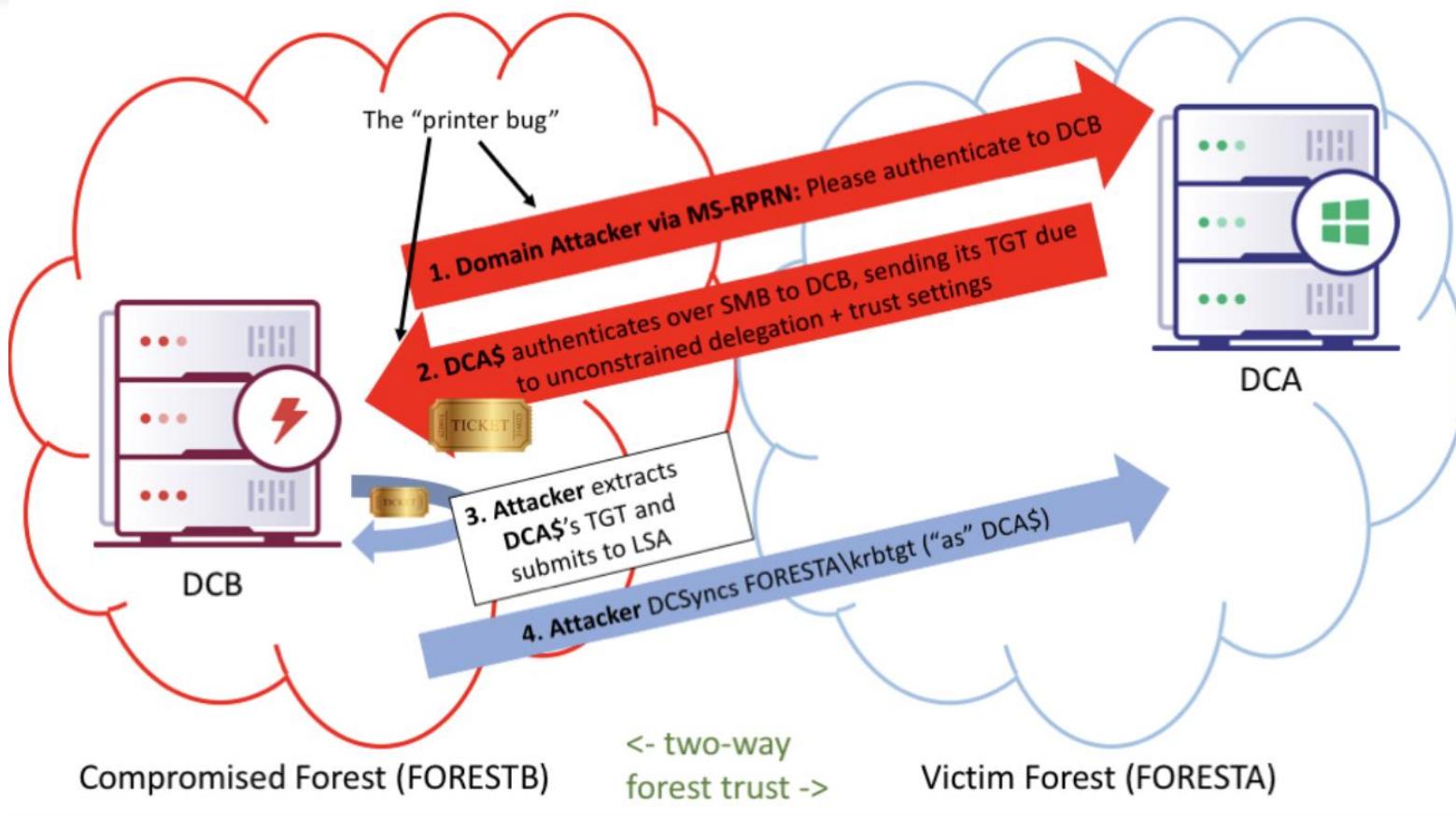
Linux Vm Size ⓘ

17763.1457.2009030514

Domain FQDN ⓘ

latest

Ejemplos? Por que es complicado?



Ejemplos? Por que es complicado?

Dirk-jan
 @_dirkjan

New blog: A different way of abusing Zerologon. No more password reset needed: using the printer bug with Zerologon to relay to DRSUAPI and DC Sync directly with ntlmrelayx: dirkjanm.io/a-different-way-of-abusing-zerologon
Code: github.com/dirkjanm/CVE-2020-0796

The diagram illustrates a four-step attack flow:

- ① The Attacker initiates an "Auth printing" session with DC1.
- ② DC1 performs an "Auth" challenge to the Attacker.
- ③ The Attacker performs a "zerologon" attack on DC2.
- ④ DC2 performs a "Relay" response to the Attacker.

3:33 PM · Sep 24, 2020 · Twitter Web App

https://twitter.com/_dirkjan/status/1309214379003588608

@Mordor_Project

- Eventos de seguridad pre-grabados, generados a través de la simulación de técnicas usadas por adversarios en
- Formato JavaScript Object Notation (**JSON**)
- Sets de datos categorizados por plataformas, grupos de adversarios, tácticas y técnicas definidas por MITRE - ATT&CK
- Datasets pequeñas y grandes



<https://mordordatasets.com>



The Mordor Project

Q Search this book...

MORDOR ENVIRONMENTS

The Shire
Erebor

HOW TO

Create Mordor Datasets
Consume Mordor Datasets

EVENTS

Mordor Events!

SMALL MORDOR DATASETS

Windows

Execution

Covenant Grunt Msbuild
Empire Invoke PSRemoting
Empire Invoke WMI Debugger
Empire Invoke WMI
Empire DCOM ShellWindows
WMIC Add User Backdoor
WMI Event Subscription
Empire Invoke PsExec
Empire Invoke Msbuild
Empire Launcher VBS
Covenant InstallUtil

←

Windows

ATT&CK Navigator View

Contents

ATT&CK Navigator View

Table View

MITRE ATT&CK® Navigator

Initial Access 9 techniques

Execution 10 techniques

Persistence 17 techniques

Privilege Escalation 12 techniques

Defense Evasion 32 techniques

Credential Access 13 techniques

Discovery 22 techniques

Access Token Manipulation (highlighted)

Drive-by Compromise

Exploit Public-Facing Application

External Remote Services

Hardware Additions

Phishing

Replication Through Removable Media

Supply Chain

Command and Scripting Interpreter

BITS Jobs

Inter-Process Communication

Native API

Scheduled Task/Job

Shared Modules

Software Deployment Tools

System

Account Manipulation

Boot or Logon Autostart Execution

Boot or Logon Initialization Scripts

Browser Extensions

Compromise Client Software Binary

Create

Abuse Elevation Control Mechanism

Access Token Manipulation

Boot or Logon Autostart Execution

BITS Jobs

Deobfuscate/Decode Files or Information

Boot or Logon Initialization Scripts

Create or Modify System Process

Event Triggered Execution

File and Direct

Abuse Elevation Control Mechanism

Access Token Manipulation

BITS Jobs

Deobfuscate/Decode Files or Information

Direct Volume Access

Execution Guardrails

Exploitation for Defense Evasion

File and Direct

Forced Authentication

Input Capture

Man-in-the-Middle

File and Direct

Brute Force

Credentials from Password Stores

Exploitation for Credential Access

Forced Authentication

Input Capture

Man-in-the-Middle

File and Direct

Network Service Scanning

Network Share

Account Discovery

Application WMI Discovery

Browser Bookmarks Discovery

Domain Trust Discovery

File and Direct Discovery

Network Share

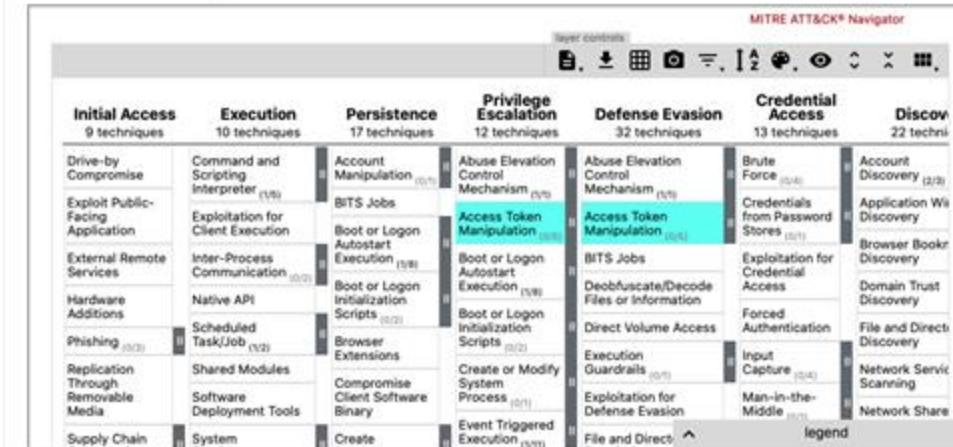


Table View

Created	Dataset	Description
2020/08/05	Covenant DCSync	This dataset represents adversaries with enough permissions (domain admin) adding an ACL to the Root Domain for any user, despite being in no privileged groups, having no malicious sidHistory, and not having local admin rights on the domain controller itself.

Técnicas específicas
(Small datasets)

Varias técnicas
relacionadas
(Large datasets)

SCM and DLL Hijack IKEEXT

Persistence

- Empire Userland Registry Run Key
- Empire Userland Scheduled Tasks
- Empire Elevated WMI Subscription
- Empire Elevated Scheduled Tasks
- SCM and DLL Hijacking IKEEXT**
- Empire Elevated Registry
- Privilege Escalation
- Empire UAC Shell API FodHelper
- Empire Invoke Runas
- Empire Elevated WMI Subscription
- Empire DLL Injection
- Empire PSInject
- SCM and DLL Hijacking IKEEXT
- Defense Evasion
- Covenant Grunt Msbuild
- Empire Wdigest Downgrade
- Empire Enabling RDP
- Empire Invoke Runas
- Empire Mimikatz OPTH
- Empire DLL Injection
- Empire Launcher SCT Regsvr32
- Empire PSInject
- SCM and DLL Hijacking IKEEXT
- Empire Invoke Msbuild
- Empire DCSync ACL
- Extended NetNTLM Downgrade
- Covenant InstallUtil
- Credential Access



Contents

Metadata

Dataset Description

Adversary View

Explore Mordor Dataset

Adversary View

```
(Empire: NZB6SE34) > upload /tmp/wlbsctrl.dll
[*] Tasked agent to upload wlbsctrl.dll, 124 KB
[*] Tasked NZB6SE34 to run TASK_UPLOAD
[*] Agent NZB6SE34 tasked with task ID 46
(Empire: NZB6SE34) > shell COPY .\wlbsctrl.dll \\HR001\C$\Windows\System32\wlbsctrl.dll
[*] Tasked NZB6SE34 to run TASK_SHELL
[*] Agent NZB6SE34 tasked with task ID 47
(Empire: NZB6SE34) > ..Command execution completed.

(Empire: NZB6SE34) > shell sc.exe `\\HR001 stop IKEEXT
[*] Tasked NZB6SE34 to run TASK_SHELL
[*] Agent NZB6SE34 tasked with task ID 48
(Empire: NZB6SE34) > SERVICE_NAME: IKEEXT
    TYPE          : 30  WIN32
    STATE         : 3  STOP_PENDING
                  (NOT_STOPPABLE, NOT_PAUSABLE, IGNORES_SHUTDOWN)
    WIN32_EXIT_CODE   : 0  (0x0)
    SERVICE_EXIT_CODE : 0  (0x0)
    CHECKPOINT      : 0x0
    WAIT_HINT       : 0x1388
..Command execution completed.

(Empire: NZB6SE34) > shell sc.exe `\\HR001 query IKEEXT
[*] Tasked NZB6SE34 to run TASK_SHELL
[*] Agent NZB6SE34 tasked with task ID 49
(Empire: NZB6SE34) > SERVICE_NAME: IKEEXT
    TYPE          : 20  WIN32_SHARE_PROCESS
    STATE         : 1  STOPPED
    WIN32_EXIT_CODE   : 0  (0x0)
    SERVICE_EXIT_CODE : 0  (0x0)
    CHECKPOINT      : 0x0
    WAIT_HINT       : 0x0
..Command execution completed.
```

Mordor: Septiembre 25 @15:30-16:30

ENTRANDO A
MORDOR
Click to add text
COMPARTIENDO DATOS PARA LA INVESTIGACIÓN DE
AMENAZAS CON LA COMUNIDAD

J. RODRIGUEZ
@CYB3RPANDAH

R. RODRIGUEZ
@CYB3RWARDOG

BLUESPACE

The graphic features a dark purple background with a grid pattern. At the top, the word "MORDOR" is written in large, bold, blue letters. Above it, smaller text reads "ENTRANDO A" and "Click to add text". Below "MORDOR", there is a subtitle in Spanish: "COMPARTIENDO DATOS PARA LA INVESTIGACIÓN DE AMENAZAS CON LA COMUNIDAD". In the bottom left corner, there is a circular portrait of a man with short hair, identified as "J. RODRIGUEZ" and "@CYB3RPANDAH". In the bottom right corner, there is another circular portrait of a man with longer hair, identified as "R. RODRIGUEZ" and "@CYB3RWARDOG". Between the two portraits is a logo for "BLUESPACE", which features a stylized blue hooded figure with a flame at the bottom.

Qué podemos hacer con la data?



Mordor Files

Entrenamiento de analistas de seguridad

Entrevistas de trabajo

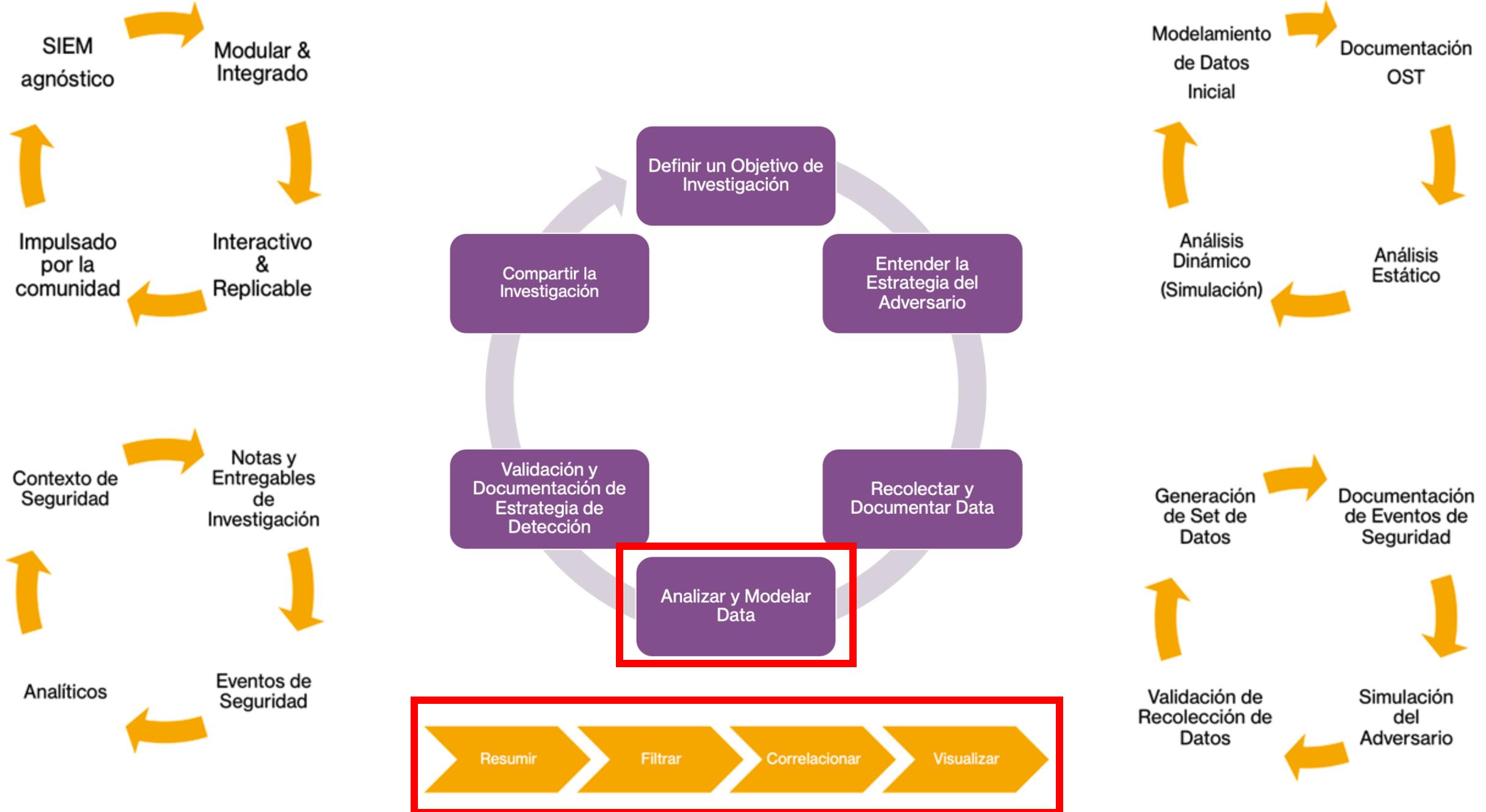
Eventos (Hackathons)

Validación de analíticos desarrollados in-house

OSSEM (Open Source Security Event Metadata)

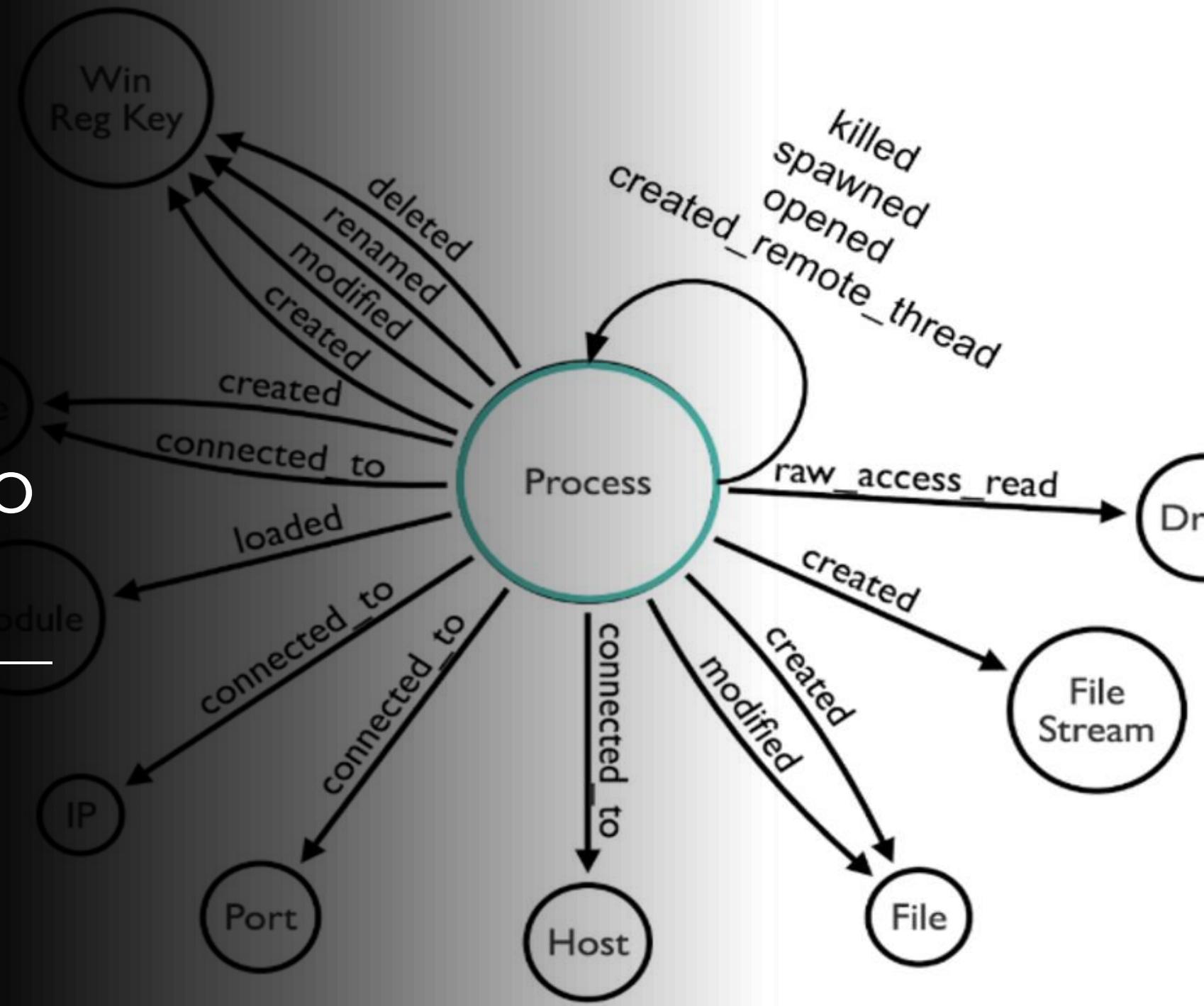
- Diccionarios de Datos
 - Windows (Security, Sysmon)
- Modelos de Datos
 - Elementos de datos
 - Relaciones
- Modeo de Informacion Común
 - Standarización





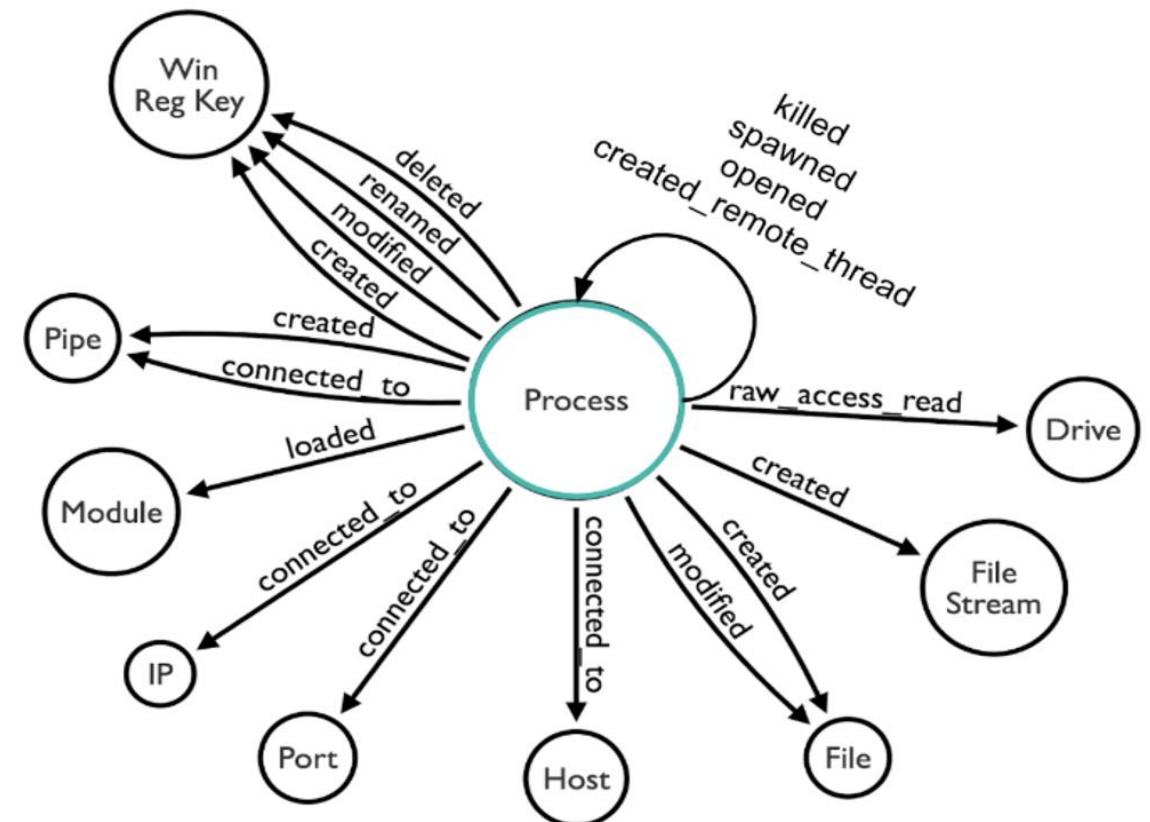
Análisis y Modelamiento de Datos

Representando al Adversario



Modelamiento de Datos

Un modelo de datos nos permite **organizar elementos de datos** que consideramos relevantes para nuestro análisis y estandarizar **la relación que existen entre ellos**



¿Cómo identificar elementos de datos y relaciones?

Un **diccionario de datos** describe un evento de seguridad y los atributos de datos que lo describen



Event Properties - Event 1, Sysmon

General Details

Process Create:
UtcTime: 2018-04-11 05:25:02.955
ProcessGuid: {a98268c1-9c2e-5acd-0000-0010396cab00}
ProcessId: 4756
Image: C:\Windows\System32\conhost.exe
FileVersion: 10.0.16299.15 (WinBuild.160101.0800)
Description: Console Window Host
Product: Microsoft® Windows® Operating System
Company: Microsoft Corporation
CommandLine: \?\C:\WINDOWS\system32\conhost.exe 0xffffffff -ForceV1
CurrentDirectory: C:\WINDOWS
User: DESKTOP-WARDOG\wardog
LogonGuid: {a98268c1-95f2-5acd-0000-002019620f00}
LogonId: 0xF6219
TerminalSessionId: 1
IntegrityLevel: Medium
Hashes: SHA1=B0BF5AC2E81BBF597FAD5F349FEEB32CAC449FA2,MD5=6A255BEBF3DBC013585538ED47DBAFD7,SHA256=4668BB223FFB983A5F1273B9E3D9FA2C5CE4A0F1FB18CA5C1B285762020073C,IMPHASH=2505BD03D7BD285E50CE89CEC02B333B
ParentProcessGuid: {a98268c1-9c2e-5acd-0000-00100266ab00}
ParentProcessId: 240
ParentImage: C:\Windows\System32\cmd.exe
ParentCommandLine: "C:\WINDOWS\system32\cmd.exe"

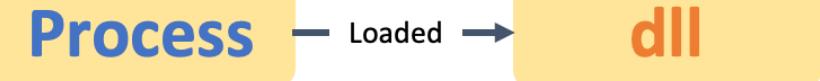
Log Name: Microsoft-Windows-Sysmon/Operational
Source: Sysmon
Event ID: 1
Level: Information
User: SYSTEM
OpCode: Info
Logged: 4/11/2018 1:25:02 AM
Task Category: Process Create (rule: ProcessCreate)
Keywords:
Computer: DESKTOP-WARDOG
More Information: [Event Log Online Help](#)

Documentando eventos de seguridad

Sysmon 7
Image Loaded

Field	Type	Description	Sample Value
Process Guid	String	Process Guid of the process that loaded the image	{A98268C1-A12A-5ACD-0000-0010E4C8B300}
Process Id	Integer	Process ID used by the os to identify the process that loaded the image	3532
Image	String	File path of the process that loaded the image	C:\Windows\System32\cmd.exe
Image Loaded	String	Full path of the image loaded	C:\Windows\System32\msvcrt.dll
Description	String	Description of the image loaded	Windows NT CRT DLL

Module



Ejemplo: Modelando el uso de COR_PROFILER

Home > Techniques > Enterprise > Hijack Execution Flow > COR_PROFILER

Hijack Execution Flow: COR_PROFILER

Other sub-techniques of Hijack Execution Flow (11)

Adversaries may leverage the COR_PROFILER environment variable to hijack the execution flow of programs that load the .NET CLR. The COR_PROFILER is a .NET Framework feature which allows developers to specify an unmanaged (or external of .NET) profiling DLL to be loaded into each .NET process that loads the Common Language Runtime (CLR). These profilers are designed to monitor, troubleshoot, and debug managed code executed by the .NET CLR.^{[1][2]}

The COR_PROFILER environment variable can be set at various scopes (system, user, or process) resulting in different levels of influence. System and user-wide environment variable scopes are specified in the Registry, where a Component Object Model (COM) object can be registered as a profiler DLL. A process scope COR_PROFILER can also be created in-memory without modifying the Registry. Starting with .NET Framework 4, the profiling DLL does not need to be registered as long as the location of the DLL is specified in the COR_PROFILER_PATH environment variable.^[2]

Adversaries may abuse COR_PROFILER to establish persistence that executes a malicious DLL in the context of all .NET processes every time the CLR is invoked. The COR_PROFILER can also be used to elevate privileges (ex: [Bypass User Access Control](#)) if the victim .NET process executes at a higher permission level, as well as to hook and [Impair Defenses](#) provided by .NET processes.^{[3][4][5][6][7]}

ID: T1574.012

Sub-technique of: [T1574](#)

Tactics: Persistence, Privilege Escalation, Defense Evasion

Platforms: Windows

Permissions Required: Administrator, User

Data Sources: File monitoring, Process command-line parameters, Process monitoring, Windows Registry

Contributors: Jesse Brown, Red Canary

Version: 1.0

Created: 24 June 2020

Last Modified: 26 June 2020

[Version Permalink](#)

Ejemplo: Modelando el uso de **COR_PROFILER**



Adversario habilita
COR_ENABLE_PROFILING variable

Adversario configura el CLSID de la
variable **COR_PROFILER**, el cual
identifica la ubicacion de la DLL

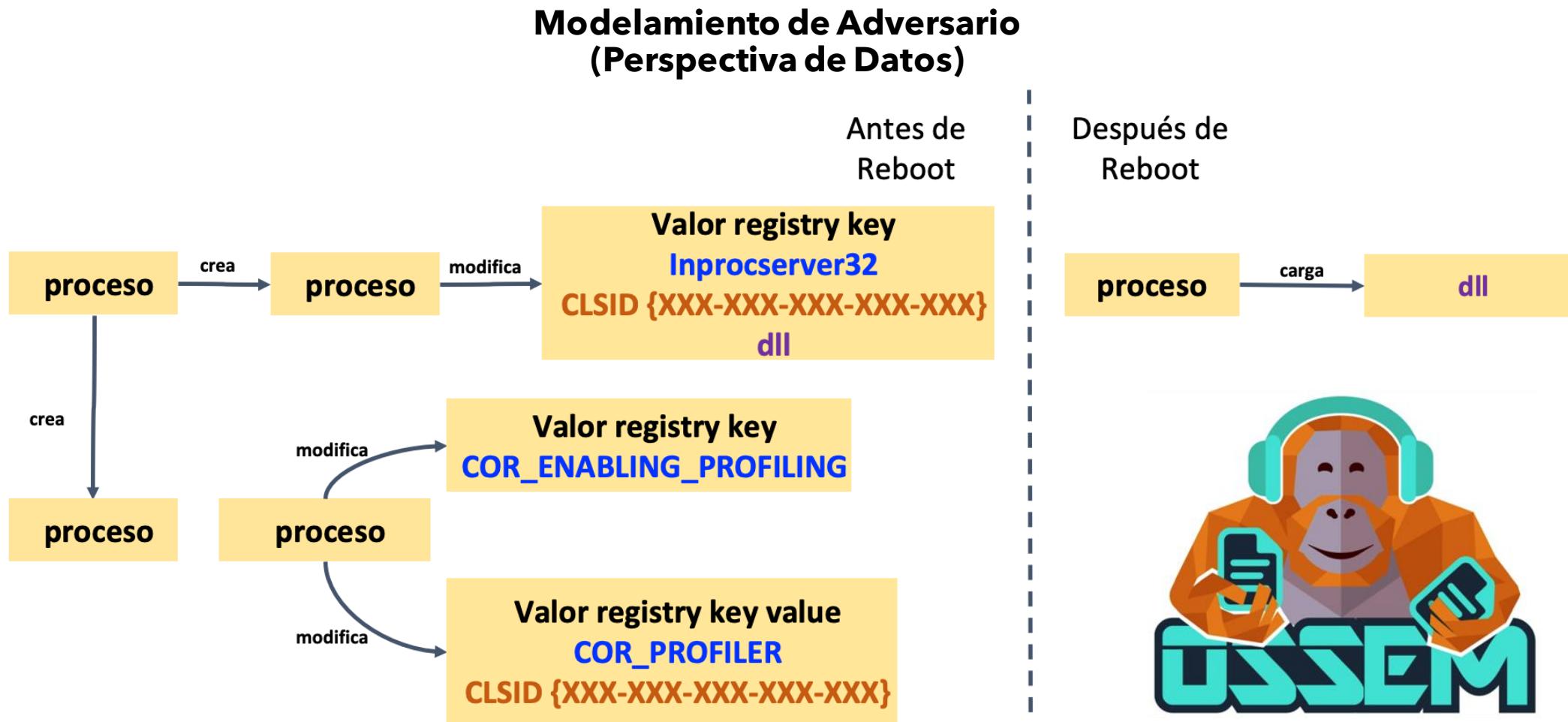
Adversario registra la DLL usando la
InProcServer32 sunkey en el registry y
asociandola al mismo CLSID

```
wmic ENVIRONMENT create  
name="COR_ENABLE_PROFILING",username="",VariableValue="1"
```

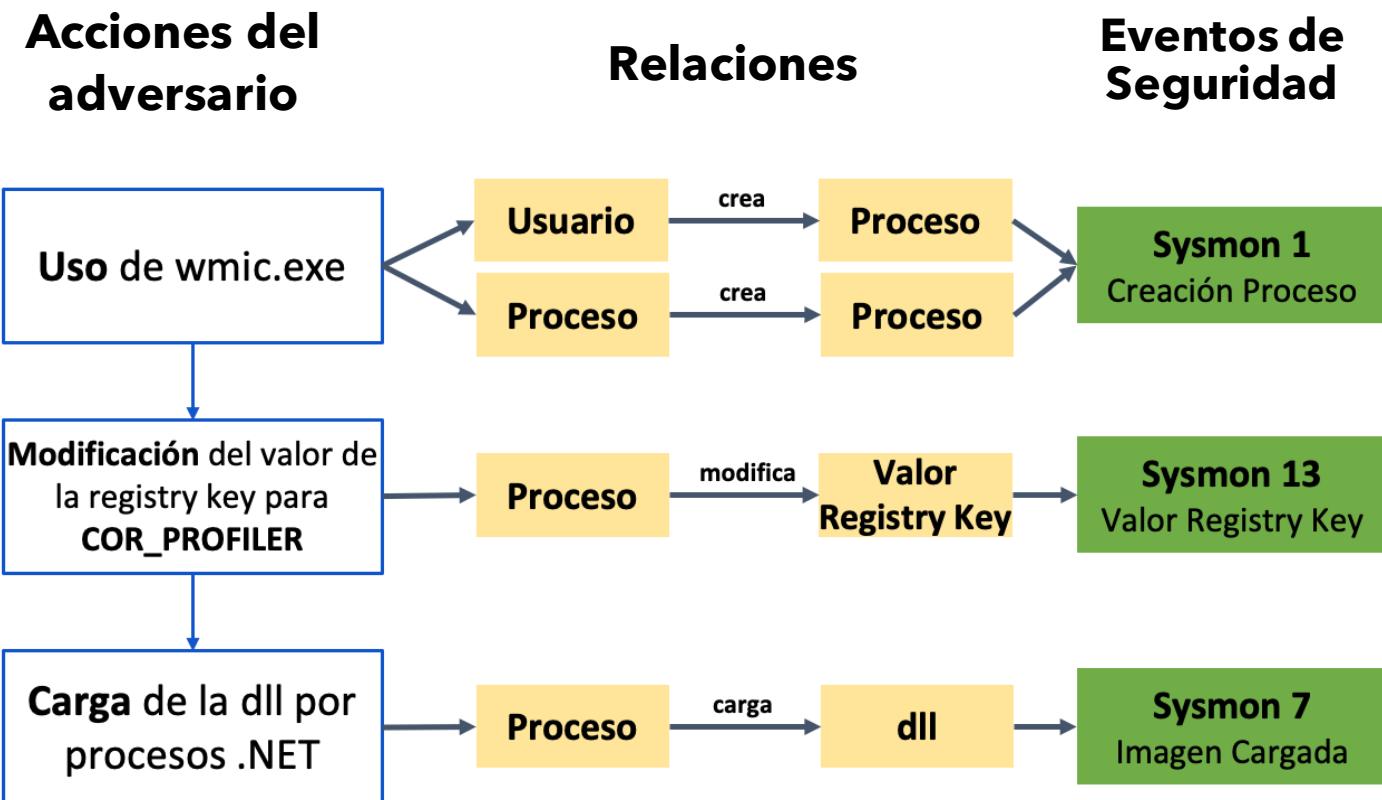
```
wmic ENVIRONMENT create  
name="COR_PROFILER",username="",VariableValue=""  
  
REG.EXE ADD HKEY_LOCAL_MACHINE\Software\Classes\CLSID\\InProcServer32  
/V ThreadingModel /T REG_SZ /D Apartment /F
```

```
REG.EXE ADD HKEY_LOCAL_MACHINE\Software\Classes\CLSID\\InProcServer32  
/VE /T REG_SZ /D "c:\windows\System32\e0b3489da74f.dll" /F
```

Ejemplo: Modelando el uso de **COR_PROFILER**



Ejemplo: Modelando el uso de **COR_PROFILER**



Jupyter Notebooks



- Limpieza y transformación de data
- Visualización de data
- Modelamiento estadístico de datos
- Aplicaciones de machine learning y mas..

¿Qué son Jupyter Notebooks?



- Son documentos que podemos accesar a través de una interface web. Nos permite gestionar y almacenar:
 - **Input:** Código (por ejemplo Python)
 - **Output:** Resultados de Código ejecutado
- Excelente para contar la historia de la investigacion desarrollada.

Python Interpreter -> IPython -> Jupyter

```
[Robertos-MacBook-Pro:~ wardog$ python3
Python 3.7.2 (default, Feb 12 2019, 08:16:38)
[Clang 10.0.0 (clang-1000.11.45.5)] on darwin
Type "help", "copyright", "credits" or "license" for more information.
[>>> print('Hola Python!!!')
Hola Python!!
[>>> 12 * 2
24
>>> ]
```

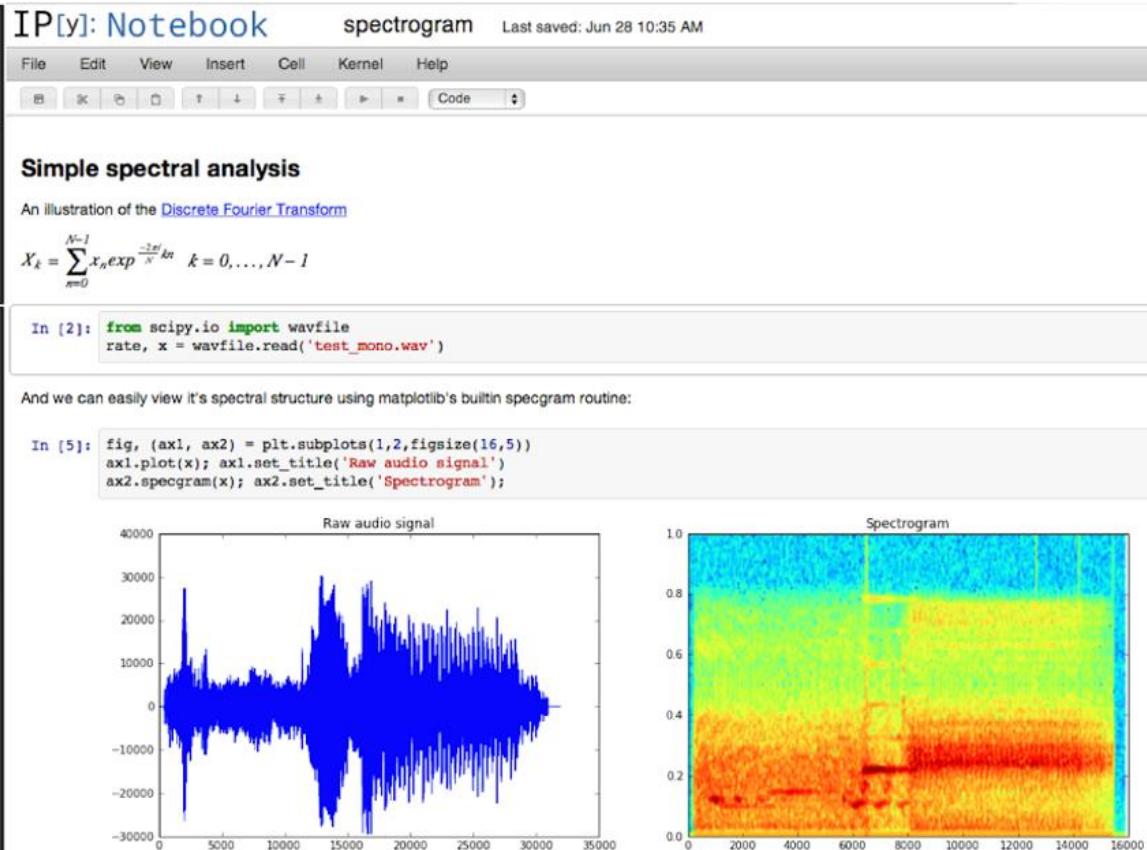
```
[Robertos-MacBook-Pro:GitHub wardog$ ipython
Python 2.7.10 (default, Oct  6 2017, 22:29:07)
Type "copyright", "credits" or "license" for more information.

IPython 5.7.0 -- An enhanced Interactive Python.
?          -- Introduction and overview of IPython's features.
%quickref -- Quick reference.
help       -- Python's own help system.
object?    -- Details about 'object', use 'object??' for extra details.

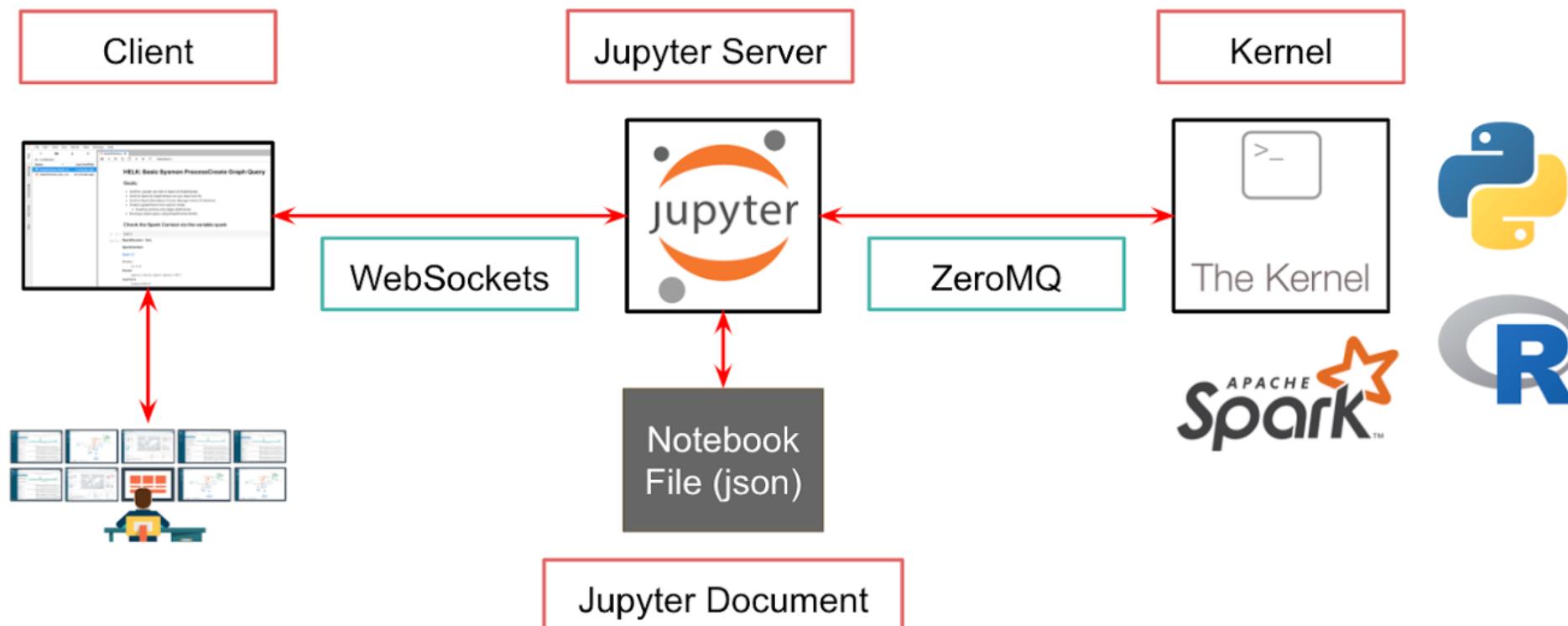
[In [1]: print('Hola IPython!!!')
Hola IPython!!

[In [2]: 12 * 2
Out[2]: 24

In [3]: ]
```



La arquitectura básica de Jupyter Notebooks



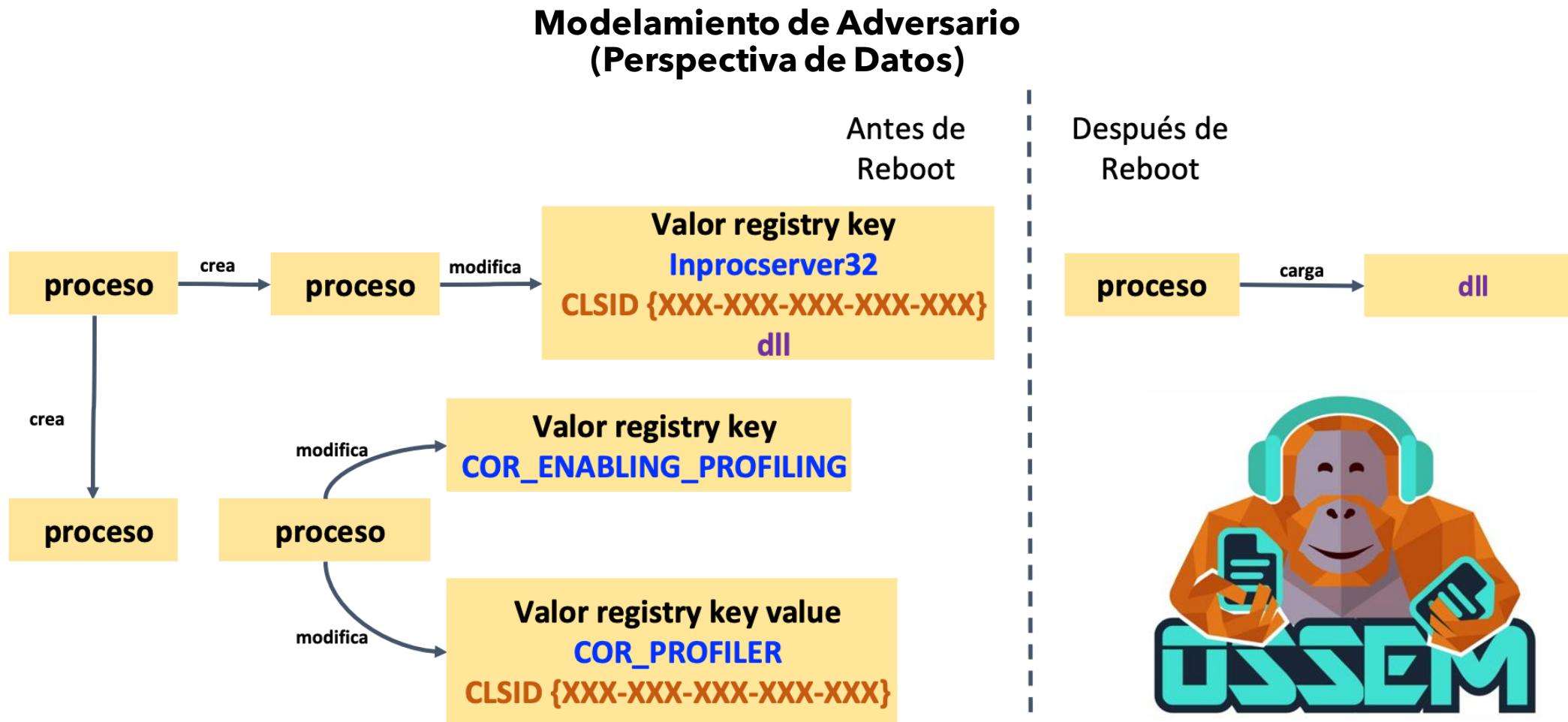
A Jupyter Notebook

The screenshot shows a Jupyter Notebook interface. At the top, there's a header bar with the Jupyter logo, the title "Untitled", a status message "Last Checkpoint: a few seconds ago (unsaved changes)", a Python 3 logo, and a "Logout" button. Below the header is a menu bar with File, Edit, View, Insert, Cell, Kernel, Widgets, and Help. To the right of the menu are buttons for Trusted and Python 3. The main area contains a code cell with the text "In [1]: print("HOLA")" and its output "HOLA". A new code cell is currently being typed, indicated by the "In []:" prompt.

```
In [1]: print("HOLA")
HOLA
```

```
In [ ]:
```

Ejemplo: Modelando el uso de **COR_PROFILER**

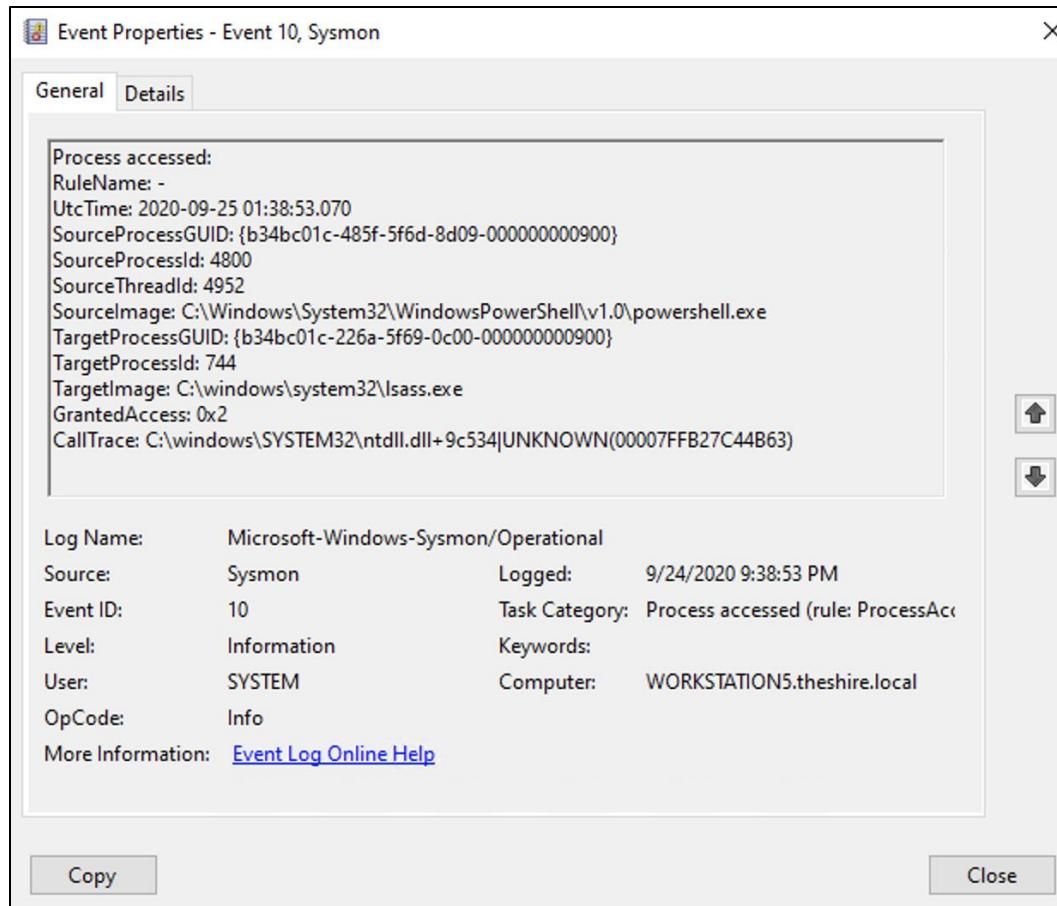


Ejemplo #1: Persistencia via COR_PROFILER

```
networkConnection = spark.sql(  
    """  
    SELECT h.Image as ProcessNetwork, h.DestinationIp, g.ProcessLoadDll, g.ProcessModifyEnvironment,  
        g.Environment, g.InprocserverProcess, g.TargetObject, g.Details  
    FROM coreProfilerExecution h  
    INNER JOIN  
        (  
            SELECT f.ProcessGuid, e.ProcessLoadDll, e.ProcessModifyEnvironment, e.Environment,  
                e.InprocserverProcess, e.TargetObject, e.Details  
            FROM coreProfilerExecution f  
            INNER JOIN  
                (  
                    SELECT d.ProcessGuid, d.Image as ProcessLoadDll, c.ProcessModifyEnvironment, c.Environment,  
                        c.InprocserverProcess, c.TargetObject, c.Details  
                    FROM coreProfilerExecution d  
                    INNER JOIN  
                        (  
                            SELECT b. Image as ProcessModifyEnvironment, b.TargetObject as Environment, b.Details as CLSID,  
                                a.Image as InprocserverProcess, a.TargetObject, a.Details  
                            FROM coreProfilerSetup b  
                            INNER JOIN  
                                (  
                                    SELECT Image, TargetObject, Details  
                                    FROM coreProfilerSetup  
                                    WHERE event_id = 13 AND lower(TargetObject) LIKE '%inprocserver%'  
                                ) a  
                            ON a.TargetObject LIKE CONCAT('%',b.Details,'%')  
                            WHERE b.event_id = 13 AND lower(b.TargetObject) LIKE '%cor_profiler%'  
                        ) c  
                    ON c.Details = d.ImageLoaded  
                    WHERE event_id = 7  
                ) e  
            ON e.ProcessGuid = f.ParentProcessGuid  
            WHERE f.event_id = 1  
        )g  
    ON g.ProcessGuid = h.ProcessGuid  
    WHERE event_id = 3  
    """)
```

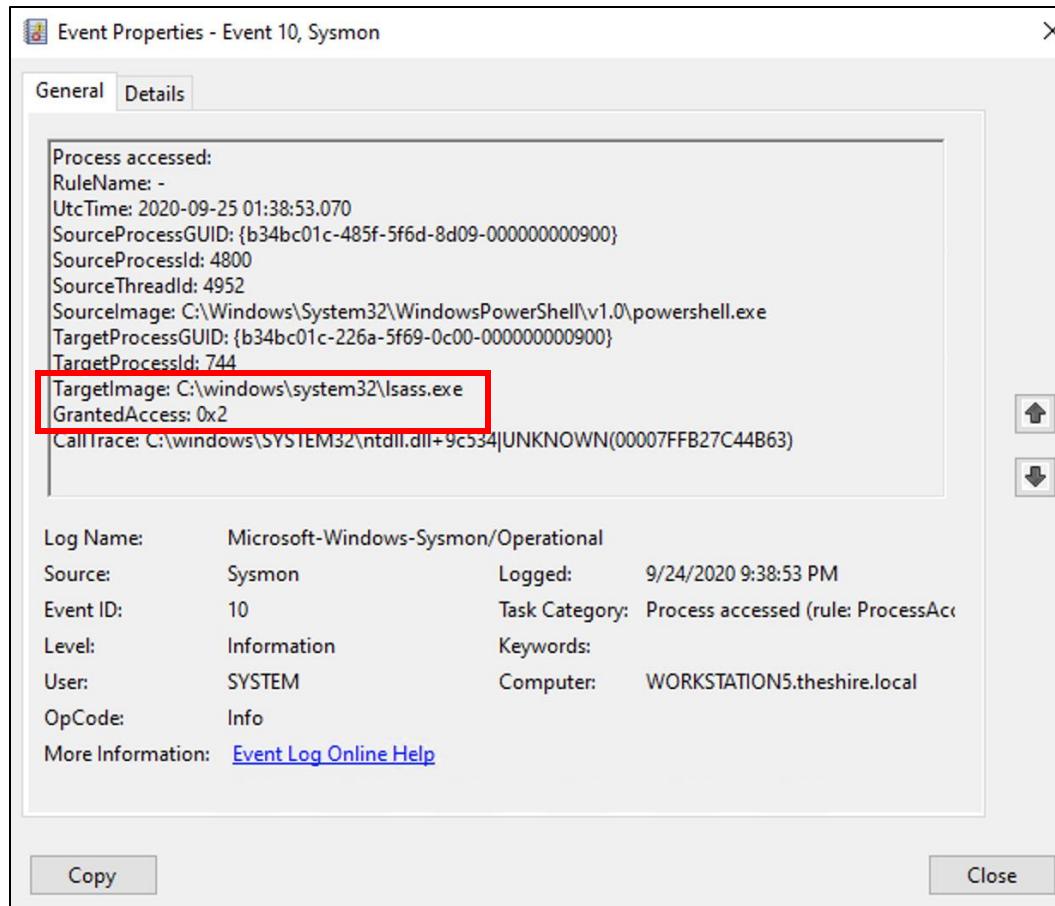
-RECORD 0-----	
ProcessNetwork	C:\Program Files (x86)\Microsoft Visual Studio\2019\Community\Common7\ServiceHub\Hosts\
ServiceHub.Host.CLR.x86\ServiceHub.SettingsHost.exe	
DestinationIp	23.217.158.90
ProcessLoadDll	C:\Program Files (x86)\Microsoft Visual Studio\2019\Community\Common7\ServiceHub\contro
ller\Microsoft.ServiceHub.Controller.exe	
ProcessModifyEnvironment	C:\WINDOWS\system32\wbem\wmiprvse.exe
Environment	HKU\\$-1-5-21-579568221-2751777276-1469956634-1001\Environment\COR_PROFILER
InprocserverProcess	C:\WINDOWS\system32\reg.exe
TargetObject	HKU\\$-1-5-21-579568221-2751777276-1469956634-1001_Classes\CLSID\{11111111-1111-1111-1111-1111deadbeef}\InProcServer32\(Default)
Details	C:\Users\jamie\Desktop\e0b3489da74f.dll
-RECORD 1-----	
ProcessNetwork	C:\Program Files (x86)\Microsoft Visual Studio\2019\Community\Common7\ServiceHub\Hosts\
ServiceHub.Host.CLR.x86\ServiceHub.IdentityHost.exe	
DestinationIp	40.126.2.2
ProcessLoadDll	C:\Program Files (x86)\Microsoft Visual Studio\2019\Community\Common7\ServiceHub\contro
ller\Microsoft.ServiceHub.Controller.exe	
ProcessModifyEnvironment	C:\WINDOWS\system32\wbem\wmiprvse.exe
Environment	HKU\\$-1-5-21-579568221-2751777276-1469956634-1001\Environment\COR_PROFILER
InprocserverProcess	C:\WINDOWS\system32\reg.exe
TargetObject	HKU\\$-1-5-21-579568221-2751777276-1469956634-1001_Classes\CLSID\{11111111-1111-1111-1111-1111deadbeef}\InProcServer32\(Default)
Details	C:\Users\jamie\Desktop\e0b3489da74f.dll
-RECORD 2-----	
ProcessNetwork	C:\Windows\System32\regsvr32.exe
DestinationIp	67.199.248.10
ProcessLoadDll	C:\Program Files (x86)\Microsoft Visual Studio\2019\Community\Common7\ServiceHub\contro
ller\Microsoft.ServiceHub.Controller.exe	
ProcessModifyEnvironment	C:\WINDOWS\system32\wbem\wmiprvse.exe
Environment	HKU\\$-1-5-21-579568221-2751777276-1469956634-1001\Environment\COR_PROFILER
InprocserverProcess	C:\WINDOWS\system32\reg.exe
TargetObject	HKU\\$-1-5-21-579568221-2751777276-1469956634-1001_Classes\CLSID\{11111111-1111-1111-1111-1111deadbeef}\InProcServer32\(Default)
Details	C:\Users\jamie\Desktop\e0b3489da74f.dll

Ejemplo #2: Accesso a LSASS



PROCESS_CREATE_PROCESS (0x0080)	Required to create a process.
PROCESS_CREATE_THREAD (0x0002)	Required to create a thread.
PROCESS_DUP_HANDLE (0x0040)	Required to duplicate a handle using DuplicateHandle .
PROCESS_QUERY_INFORMATION (0x0400)	Required to retrieve certain information about a process, such as its token, exit code, and priority class (see OpenProcessToken).
PROCESS_QUERY_LIMITED_INFORMATION (0x1000)	Required to retrieve certain information about a process (see GetExitCodeProcess , GetPriorityClass , IsProcessInJob , QueryFullProcessImageName). A handle that has the PROCESS_QUERY_INFORMATION access right is automatically granted PROCESS_QUERY_LIMITED_INFORMATION. Windows Server 2003 and Windows XP: This access right is not supported.

Ejemplo #2: Accesso a LSASS



PROCESS_CREATE_PROCESS (0x0080)	Required to create a process.
PROCESS_CREATE_THREAD (0x0002)	Required to create a thread.
PROCESS_DUP_HANDLE (0x0040)	Required to duplicate a handle using DuplicateHandle .
PROCESS_QUERY_INFORMATION (0x0400)	Required to retrieve certain information about a process, such as its token, exit code, and priority class (see OpenProcessToken).
PROCESS_QUERY_LIMITED_INFORMATION (0x1000)	Required to retrieve certain information about a process (see GetExitCodeProcess , GetPriorityClass , IsProcessInJob , QueryFullProcessImageName). A handle that has the PROCESS_QUERY_INFORMATION access right is automatically granted PROCESS_QUERY_LIMITED_INFORMATION. Windows Server 2003 and Windows XP: This access right is not supported.

Que tipos de análisis podemos desarrollar?

Create a Spark UDF to get the specific Access Rights related to every Bitmask

- Define a function

```
def getSpecificAccessRights(bitmask):
    bitmask = int(bitmask,16)
    specificAccessRights = {'PROCESS_CREATE_PROCESS' : 0x0080,
                           'PROCESS_CREATE_THREAD' : 0x0002,
                           'PROCESS_DUP_HANDLE' : 0x0040,
                           'PROCESS_QUERY_INFORMATION' : 0x0400,
                           'PROCESS_QUERY_LIMITED_INFORMATION' : 0x1000,
                           'PROCESS_SET_INFORMATION' : 0x0200,
                           'PROCESS_SET_QUOTA' : 0x0100,
                           'PROCESS_SET_SUSPEND_RESUME' : 0x0800,
                           'PROCESS_TERMINATE' : 0x0001,
                           'PROCESS_VM_OPERATION' : 0x0008,
                           'PROCESS_VM_READ' : 0x0010,
                           'PROCESS_VM_WRITE' : 0x0020,
                           'SYNCHRONIZE' : 0x00100000,
                           'PROCESS_SET_LIMITED_INFORMATION' : 0x2000}

    rights = [ ]
    for key,value in specificAccessRights.items():
        if value & bitmask != 0:
            rights.append(key)

    return rights
```

- Register Spark UDF

```
from pyspark.sql.types import *
spark.udf.register("getAccessRights", getSpecificAccessRights,ArrayType(StringType()))
```

```
<function __main__.getSpecificAccessRights(bitmask)>
```

Transformación de Datos

Que tipos de análisis podemos desarrollar?

- Apply the Spark UDF

```
processAccessRights = spark.sql(  
    """  
    SELECT GrantedAccess, getAccessRights(GrantedAccess) as RightsRequested, count(*) as Count  
    FROM processInjection  
    WHERE lower(Channel) LIKE '%sysmon%'  
        AND EventID = 10  
    GROUP BY GrantedAccess, RightsRequested  
    ORDER BY Count DESC  
    """)  
  
print('This dataframe has {} records!!'.format(processAccessRights.count()))  
processAccessRights.show(truncate = 80)
```

```
This dataframe has 10 records!!  
+-----+-----+  
|GrantedAccess| RightsRequeste  
+-----+-----+  
| 0x1000| [PROCESS_QUERY_LIMITED_INFORMATION  
| 0x3000| [PROCESS_QUERY_LIMITED_INFORMATION, PROCESS_SET_LIMITED_INFORMATION  
| 0x40| [PROCESS_DUP_HANDLE  
| 0x1400| [PROCESS_QUERY_INFORMATION, PROCESS_QUERY_LIMITED_INFORMATION  
| 0x1410| [PROCESS_QUERY_INFORMATION, PROCESS_QUERY_LIMITED_INFORMATION, PROCESS_VM_READ  
| 0x1478| [PROCESS_DUP_HANDLE, PROCESS_QUERY_INFORMATION, PROCESS_QUERY_LIMITED_INFORMATION..  
| 0xfffffff|[PROCESS_CREATE_PROCESS, PROCESS_CREATE_THREAD, PROCESS_DUP_HANDLE, PROCESS_Q..  
| 0x1f3fff|[PROCESS_CREATE_PROCESS, PROCESS_CREATE_THREAD, PROCESS_DUP_HANDLE, PROCESS_Q..  
| 0x100000| [SYNCHRONIZE  
| 0x101541|[PROCESS_DUP_HANDLE, PROCESS_QUERY_INFORMATION, PROCESS_QUERY_LIMITED_INFORMATION..  
+-----+-----+
```

**Selección y
Resumen de
Datos**

Que tipos de análisis podemos desarrollar?

Find Source Processes that used CreateRemoteThread APIs

```
networkConnection = spark.sql(  
    ...  
    "SELECT b. SourceImage, b.TargetImage, a.NewThreadId  
    FROM processInjection b  
    INNER JOIN(  
        SELECT SourceProcessGuid, NewThreadId  
        FROM processInjection  
        WHERE lower(Channel) LIKE '%sysmon%'  
        AND EventID = 8  
    )a  
    ON b.SourceProcessGUID = a.SourceProcessGuid  
    WHERE lower(Channel) LIKE '%sysmon%'  
    AND b.EventID = 10  
    AND array_contains(getAccessRights(GrantedAccess), 'PROCESS_CREATE_THREAD')  
    ...)  
  
print('This dataframe has {} records!!'.format(networkConnection.count()))  
networkConnection.show(truncate = 40)
```

```
This dataframe has 88 records!!  
+-----+-----+-----+  
|       SourceImage| TargetImage|NewThreadId|  
+-----+-----+-----+  
|C:\windows\System32\WindowsPowerShell...|C:\windows\system32\notepad.exe| 3004|  
|C:\windows\System32\WindowsPowerShell...|C:\windows\system32\notepad.exe| 3756|  
|C:\windows\System32\WindowsPowerShell...|C:\windows\system32\notepad.exe| 2836|  
|C:\windows\System32\WindowsPowerShell...|C:\windows\system32\notepad.exe| 5764|  
|C:\windows\System32\WindowsPowerShell...|C:\windows\system32\notepad.exe| 8044|  
|C:\windows\System32\WindowsPowerShell...|C:\windows\system32\notepad.exe| 6168|
```

Correlación de Datos

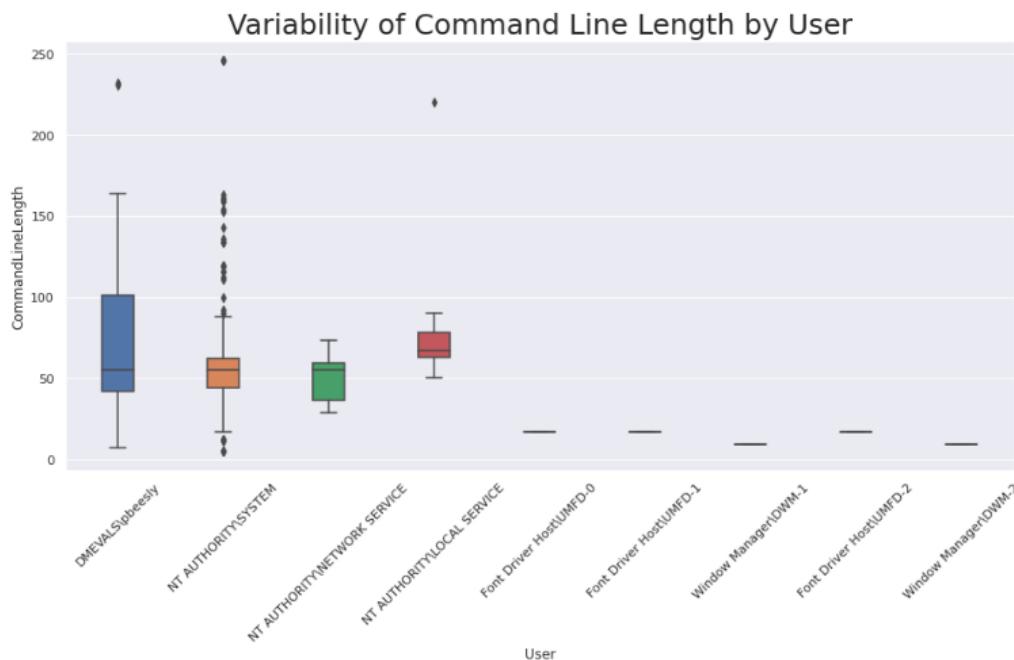
Que tipos de análisis podemos desarrollar?

```
# Source of Data
source = commandLineLength.toPandas()

# seaborn object
boxPlotChart = sns.boxplot(x = 'User', y = 'CommandLineLength', data = source, orient = 'v',width=10)

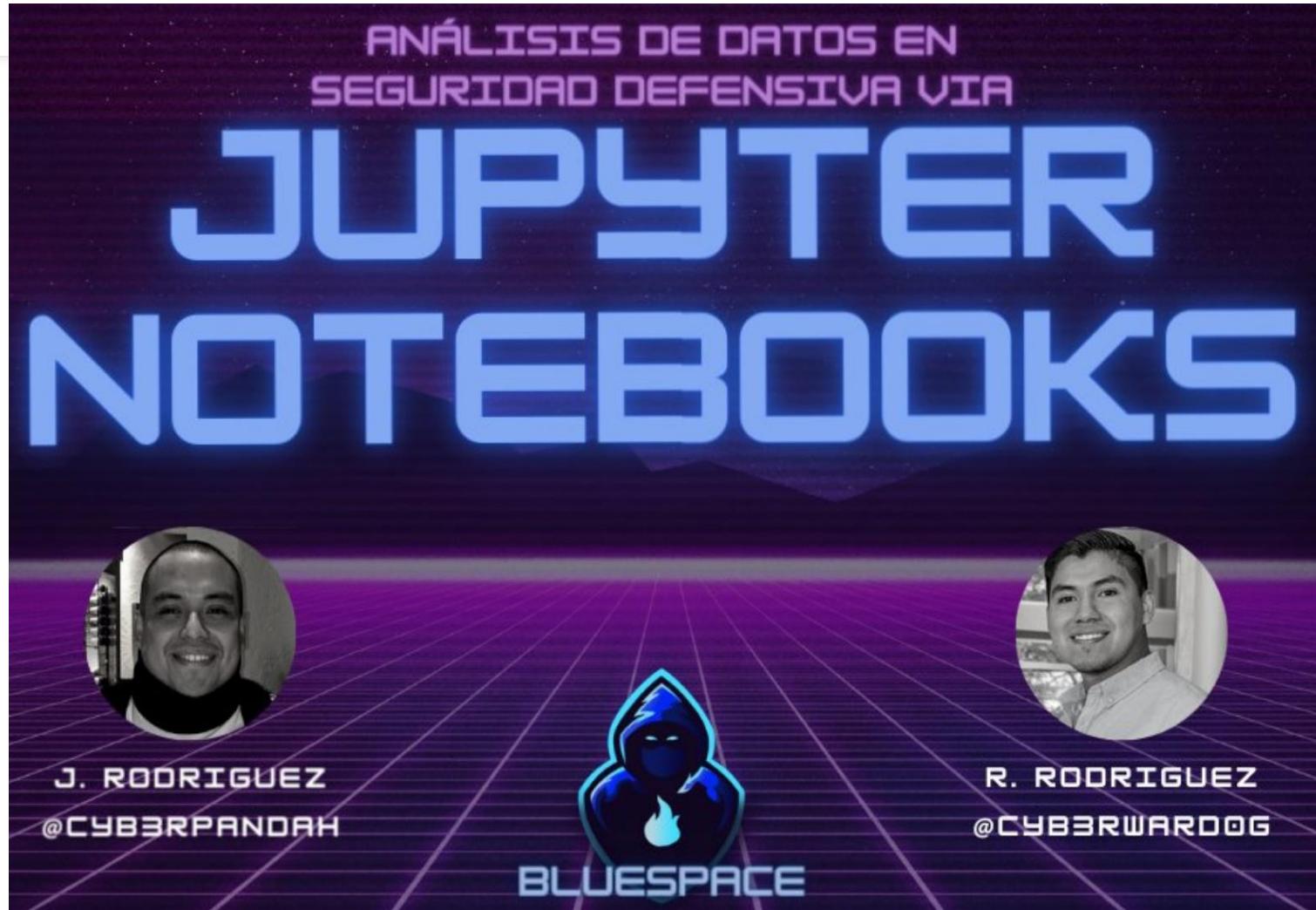
# Title format
boxPlotChart.set_title("Variability of Command Line Length by User", fontsize = 25)

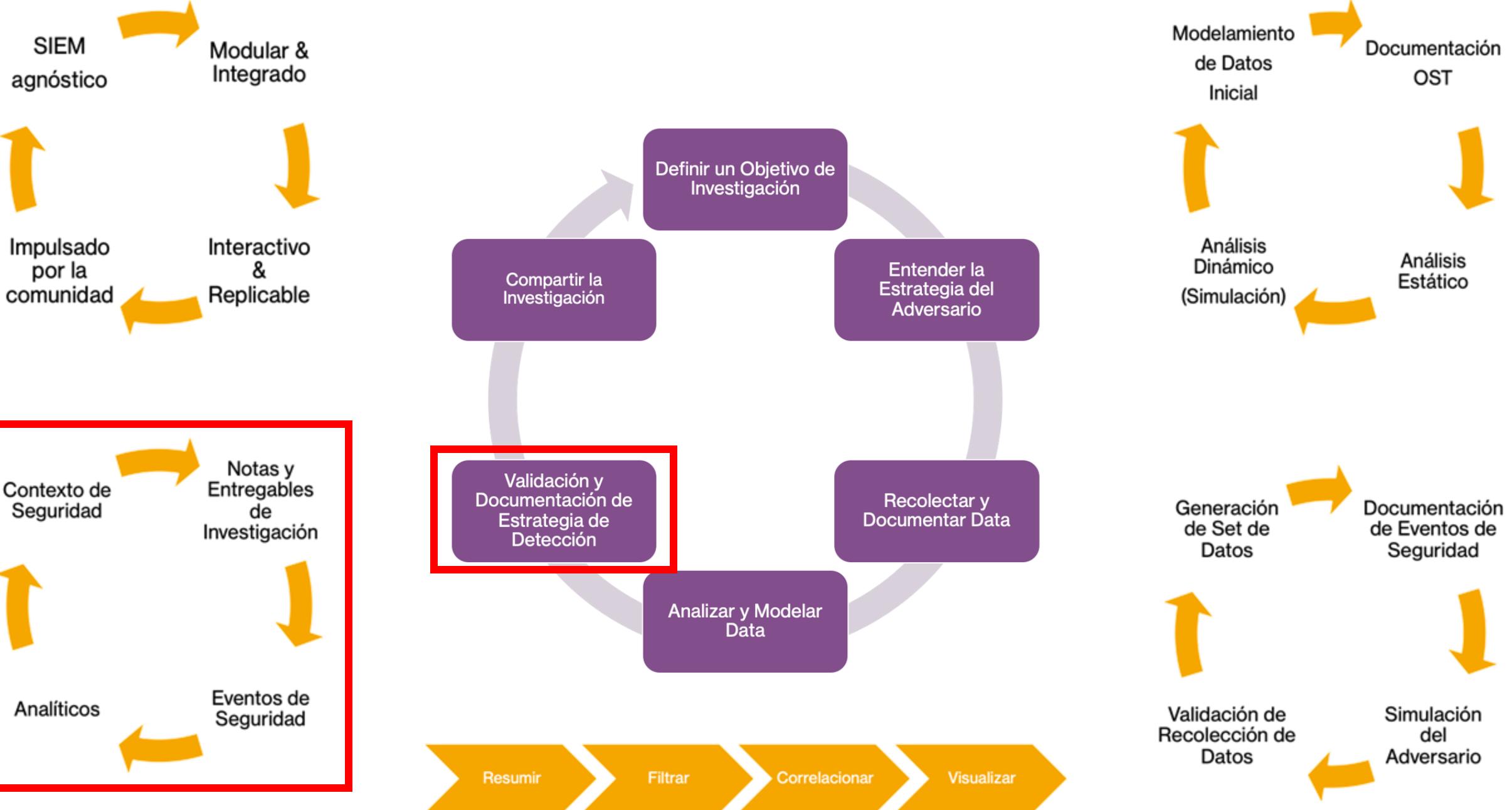
# X-axis format
boxPlotChart.set_xticklabels(boxPlotChart.get_xticklabels(), rotation=45);
```



Visualización de Datos

Jupyter Notebooks: Septiembre 26 @15:00-17:00





Validación y Documentación de Estrategia

Contando la historia detrás del resultado

Download & Process Mordor File

```
mordor_file = "https://raw.githubusercontent.com/OTRF/mordor/master/datasets/small/windows-sysmon-2017-01-01.log.gz"
spark.read().format("com.databricks.spark.csv").option("header", "true").load(mordor_file)
registerMordorSQLTable(spark, mordor_file, "mordorTable")
```

Analytic I

FP Rate	Log Channel	Description
Medium	['Microsoft-Windows-Sysmon/Operational']	Look for the creation of Event consumers type.

```
df = spark.sql(
    """
    SELECT EventID, EventType
    FROM mordorTable
    WHERE Channel = 'Microsoft-Windows-Sysmon/Operational'
        AND EventID = 20
        AND LOWER(Message) Like '%type: script%'
    """
)
df.show(10, False)
```

ThreatHunter-Playbook

- Conjunto de Notebooks que colaboran con el desarrollo de tecnicas e hipotesis en investigacion de detecciones.
- Este proyecto documenta estrategias en forma de **notebooks interactivos**, con el objetivo de proveer los resultados de la investigacion en forma sencilla y flexible.
- Cada notebooks interactúa con un set de datos del proyecto **Mordor** 😷



Playbooks estan mapeados a ATT&CK

PRE-HUNT ACTIVITIES
Data Management

CAMPAIGN NOTEBOOKS
ATT&CK Evaluations

TARGETED NOTEBOOKS
Windows

- Execution
 - Alternate PowerShell Hosts
 - WMI Win32_Process Class and Create
 - Method for Remote Execution
 - Basic PowerShell Execution
 - Service Creation
 - Alternate PowerShell Hosts
 - WMI Module Load
 - PowerShell Remote Session
 - PowerShell Remote Session
- Persistence
 - WMI Eventing
 - Remote WMI
 - ActiveScriptEventConsumers
 - Privilege Escalation
 - Remote WMI
 - ActiveScriptEventConsumers
- Defense Evasion
 - DLL Injection via CreateRemoteThread and LoadLibrary
 - Enable Remote Desktop Connections
 - Registry
 - WDigest Downgrade

←

Windows

ATT&CK Navigator View

Table View

Created	Analytic	Hypothesis	Author



Documentación y Validación de Analíticos con Sets de datos del proyecto **Mordor**

Screenshot of a Jupyter Notebook interface showing a SQL query and its results.

The sidebar contains a table of contents for the Threat Hunter Playbook, including sections like CreateRemoteThread and LoadLibrary, Registry, WDigest Downgrade, Active Directory Replication User, Backdoor, Credential Access, Domain DPAPI Backup Key Extraction, SAM Registry Hive Handle Request, Extended NetNTLM Downgrade, Active Directory Replication From, Non-Domain-Controller Accounts, Remote Interactive Task Manager, LSASS Dump, LSASS Access from Non System Account, Discovery, SAM Registry Hive Handle Request, SysKey Registry Keys Access, Remote Service Control Manager Handle, Lateral Movement, Remote Service creation, WMI Win32_Process Class and Create, Method for Remote Execution, Remote WMI, ActiveScriptEventConsumers, PowerShell Remote Session, Collection, Access to Microphone Device, and Linux.

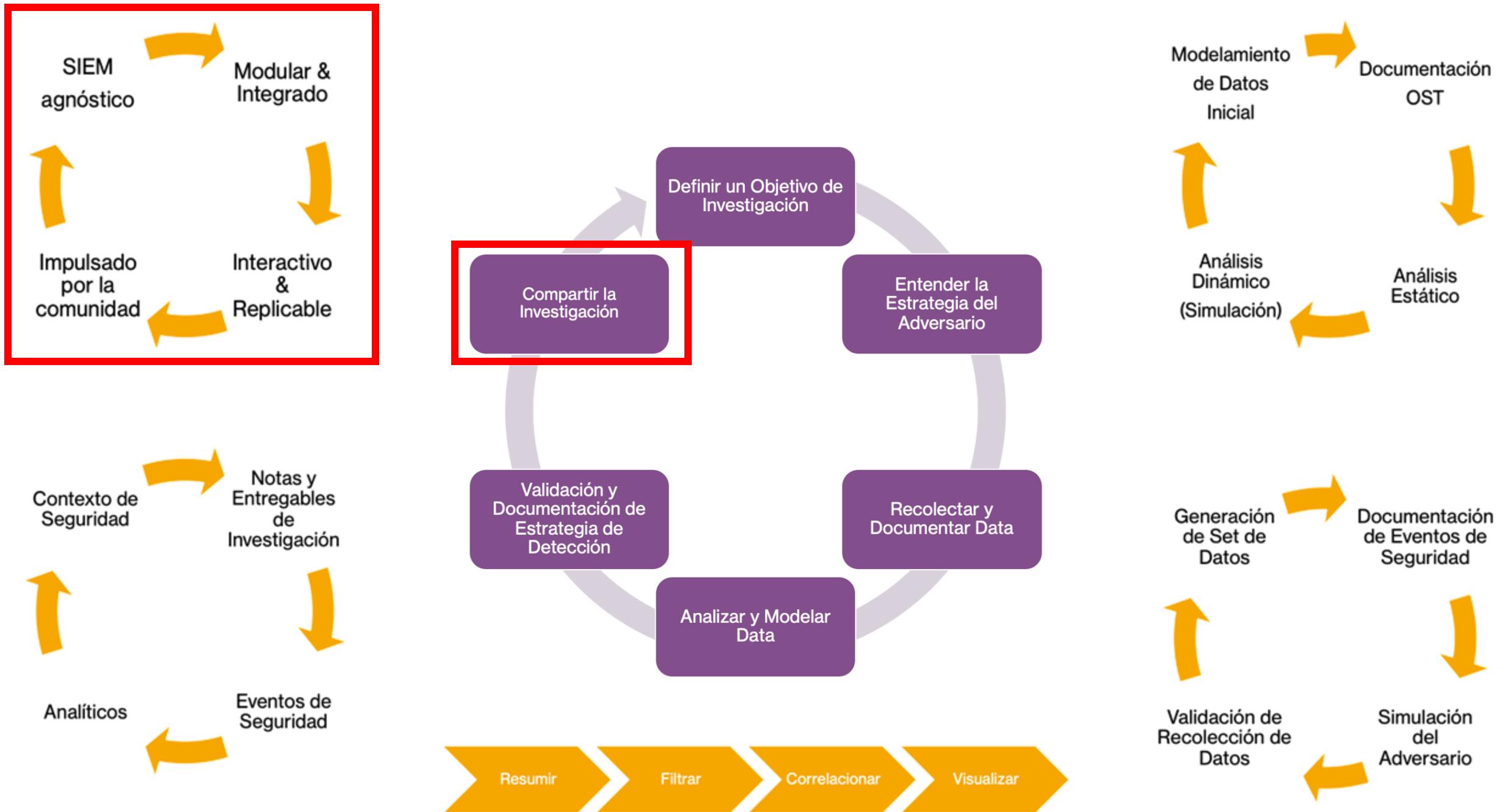
The main area shows a code cell with the following SQL query:

```
df = spark.sql(  
    ...  
    SELECT d.`#timestamp`, d.TargetUserName, c.Image, c.ProcessId  
    FROM mordorTable d  
    INNER JOIN (  
        SELECT b.ImageLoaded, a.CommandLine, b.ProcessGuid, a.Image, b.ProcessId  
        FROM mordorTable b  
        INNER JOIN (  
            SELECT ProcessGuid, CommandLine, Image  
            FROM mordorTable  
            WHERE Channel = "Microsoft-Windows-Sysmon/Operational"  
            AND EventID = 1  
            AND Image LIKE '%scrcons.exe'  
        ) a  
        ON b.ProcessGuid = a.ProcessGuid  
        WHERE b.Channel = "Microsoft-Windows-Sysmon/Operational"  
        AND b.EventID = 7  
        AND LOWER(b.ImageLoaded) IN (  
            'c:\\windows\\system32\\\\wbem\\\\scrcons.exe',  
            'c:\\windows\\system32\\\\vbscript.dll',  
            'c:\\windows\\system32\\\\wbem\\\\wbemdisp.dll',  
            'c:\\windows\\system32\\\\wshom.ocx',  
            'c:\\windows\\system32\\\\scrun.dll'  
        )  
    ) c  
    ON split(d.ProcessId, '0x')[1] = LOWER(hex(CAST(c.ProcessId as INT)))  
    WHERE LOWER(d.Channel) = "security"  
    AND d.EventID = 4624  
    AND d.LogonType = 3  
    ...  
)  
df.show(10, False)
```

Below the code cell is a preview of the resulting DataFrame:

#timestamp	TargetUserName	Image	ProcessId
2020-09-02T01:44:11.726Z	pgustavo	C:\Windows\System32\wbem\scrcons.exe	972
2020-09-02T01:44:11.726Z	pgustavo	C:\Windows\System32\wbem\scrcons.exe	972
2020-09-02T01:44:11.726Z	pgustavo	C:\Windows\System32\wbem\scrcons.exe	972
2020-09-02T01:44:11.726Z	pgustavo	C:\Windows\System32\wbem\scrcons.exe	972
2020-09-02T01:44:11.726Z	pgustavo	C:\Windows\System32\wbem\scrcons.exe	972



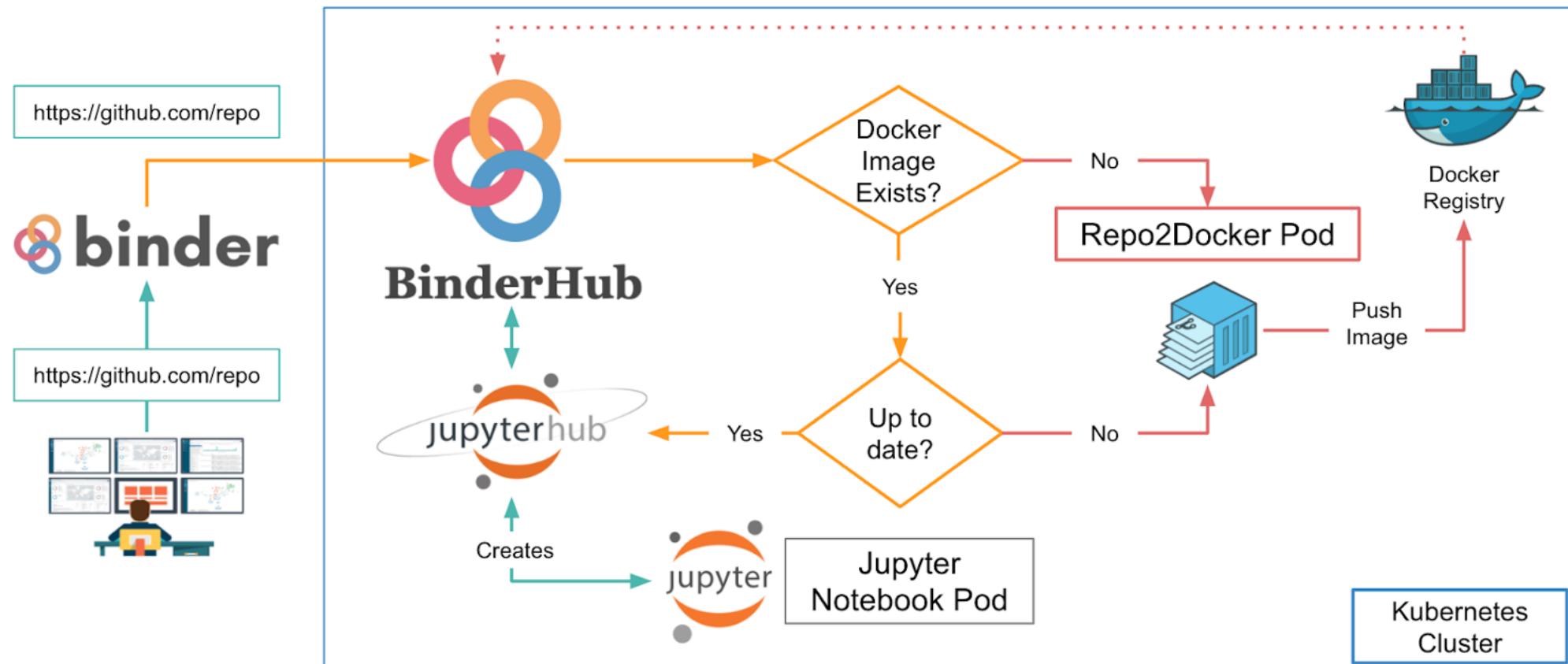


Compartiendo con la Comunidad

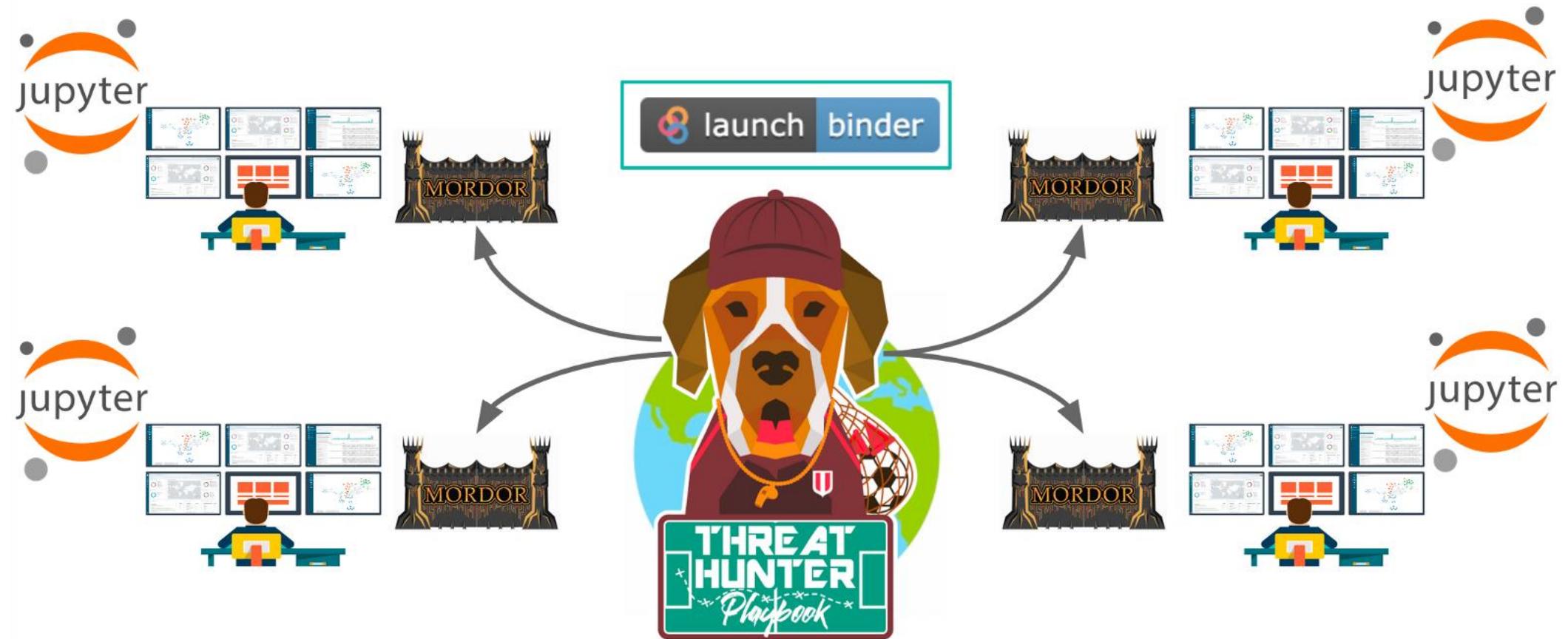
En forma práctica, interactiva y
replicable



Reponible? Práctico? Interactivo?



Replicable? Práctico? Interactivo?



Replicable? Práctico? Interactivo?

 ThreatHunter-Playbook @HunterPlaybook · 17h
"Adversaries might be leveraging WMI event subscriptions (ActiveScriptEventConsumers) for remote code execution"
@OTR_Community 🌎

📁 Playbook: threathunterplaybook.com/notebooks/wind...

😈 @Mordor_Project datasets: mordordatasets.com/notebooks/small...

📡 Reference: @domchell mdsec.co.uk/2020/09/i-like...



Remote WMI ActiveScriptEventConsumers
Remote WMI ActiveScriptEventConsumers Metadata
id WIN-200902020333 author Roberto Rodriguez ...
🔗 threathunterplaybook.com

💬 64 ⚡ 134 ⚡

Replicable? Práctico? Interactivo?

The screenshot shows a Threat Hunter Playbook page for a specific item. At the top, there's a navigation bar with icons for back, forward, search, and refresh. Below it is a header with the title "Remote WMI ActiveScriptEventConsumer". To the right of the title is a "Launch Binder" button, followed by three dark grey buttons: "Binder", "Colab", and "Live Code". Further to the right is a download icon. On the far right, there's a sidebar titled "Contents" with a list of sections: Metadata, Technical Description, Hypothesis, Analytics, Detection Blindspots, Hunter Notes, Hunt Output, and References. The "Metadata" section is currently selected and expanded. Under "Metadata", there are five entries: id (WIN-200902020333), author (Roberto Rodriguez @Cyb3rWard0g), creation date (2020/09/02), platform (Windows), and playbook link (a link that is partially cut off). The main content area below the sidebar is currently empty.

←

Remote WMI ActiveScriptEventConsumer

Metadata

id WIN-200902020333

author Roberto Rodriguez @Cyb3rWard0g

creation date 2020/09/02

platform Windows

playbook link

Binder

Colab

Live Code

Contents

Metadata

Technical Description

Hypothesis

Analytics

Detection Blindspots

Hunter Notes

Hunt Output

References

Replicable? Práctico? Interactivo?

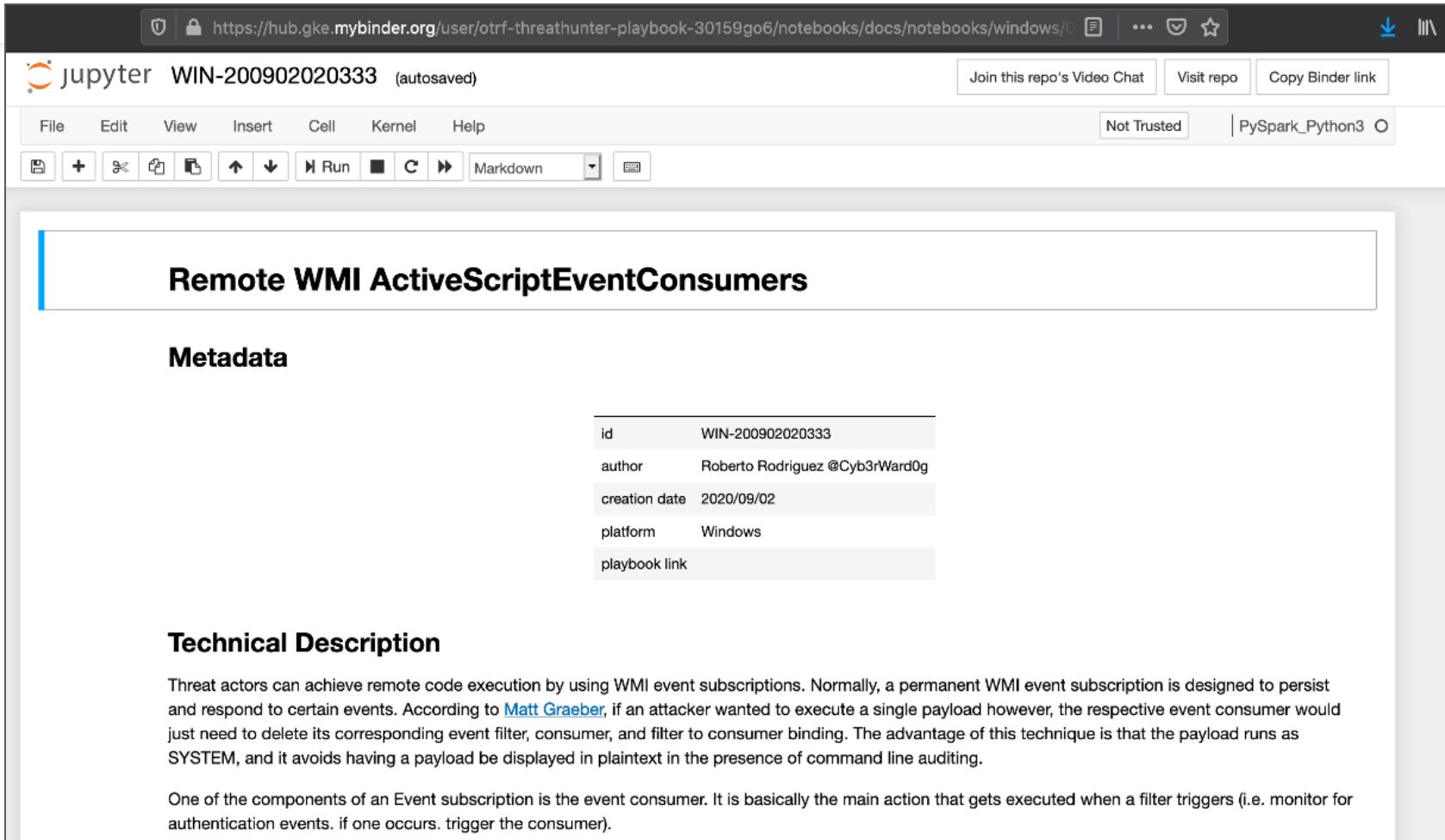
The screenshot shows a web interface for a Binder repository. At the top is the Binder logo, which consists of three overlapping circles in orange, blue, and pink. To the right of the logo is the word "binder" in a large, dark font. Below the logo is a circular loading icon with alternating orange and pink segments. The main content area contains the following text:
Starting repository: OTRF/ThreatHunter-Playbook/master
If a repository takes a long time to launch, it is usually because Binder needs
to create the environment for the first time.

Build logs

Found built image, launching...
Launching server...

hide

Replicable? Práctico? Interactivo?



The screenshot shows a Jupyter Notebook interface running on a Windows machine (WIN-200902020333). The title bar indicates the notebook is autosaved. The toolbar includes standard options like File, Edit, View, Insert, Cell, Kernel, Help, and various cell type icons. The main content area displays a section titled "Remote WMI ActiveScriptEventConsumers".

Metadata

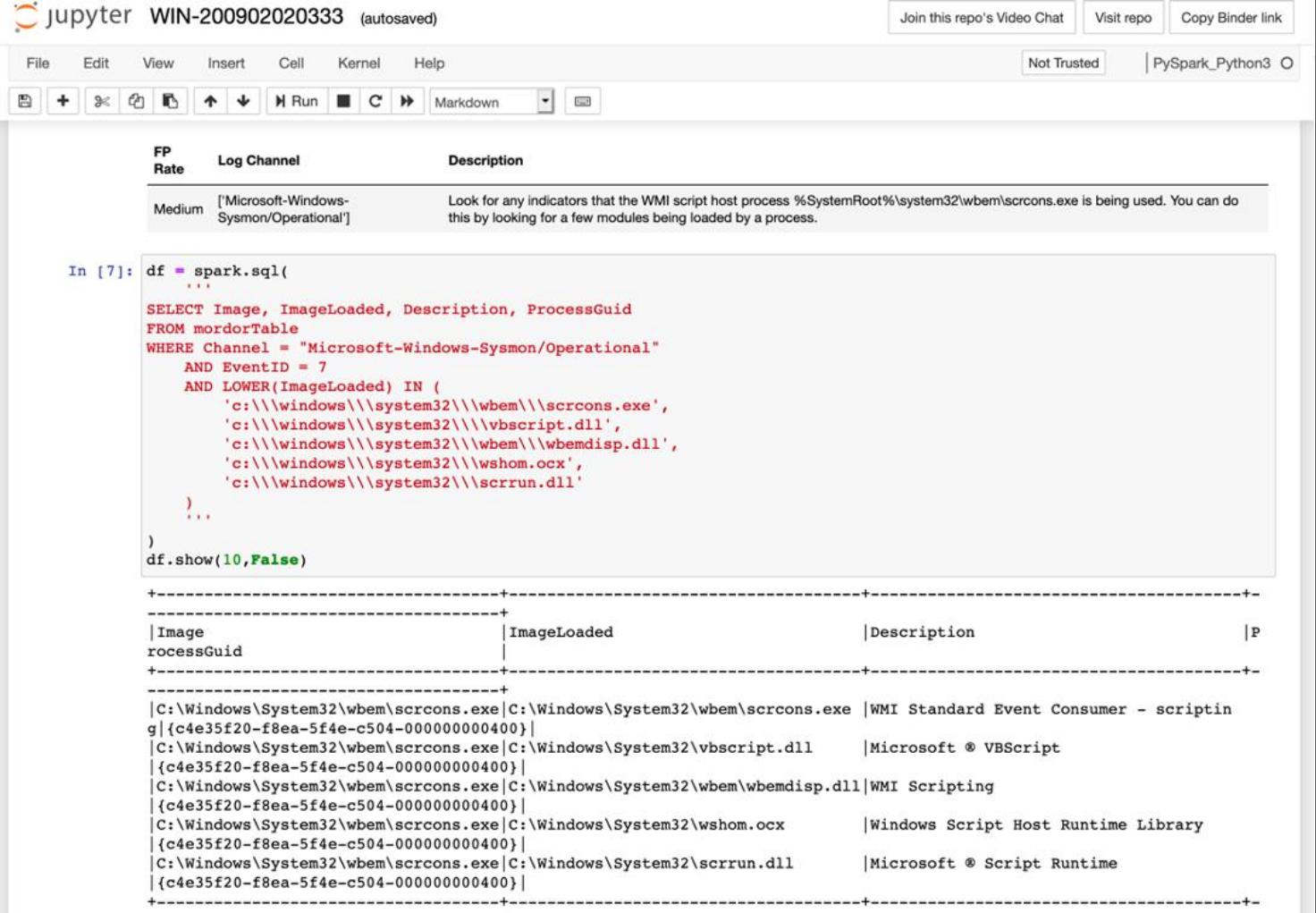
id	WIN-200902020333
author	Roberto Rodriguez @Cyb3rWard0g
creation date	2020/09/02
platform	Windows
playbook link	

Technical Description

Threat actors can achieve remote code execution by using WMI event subscriptions. Normally, a permanent WMI event subscription is designed to persist and respond to certain events. According to [Matt Graeber](#), if an attacker wanted to execute a single payload however, the respective event consumer would just need to delete its corresponding event filter, consumer, and filter to consumer binding. The advantage of this technique is that the payload runs as SYSTEM, and it avoids having a payload be displayed in plaintext in the presence of command line auditing.

One of the components of an Event subscription is the event consumer. It is basically the main action that gets executed when a filter triggers (i.e. monitor for authentication events. if one occurs. trigger the consumer).

Replicable? Práctico? Interactivo?



The screenshot shows a Jupyter Notebook interface with the title "jupyter WIN-200902020333 (autosaved)". The notebook contains a log entry and a PySpark SQL query.

Log Channel:

FP Rate	Log Channel	Description
Medium	['Microsoft-Windows-Sysmon/Operational']	Look for any indicators that the WMI script host process %SystemRoot%\system32\wbem\scrcons.exe is being used. You can do this by looking for a few modules being loaded by a process.

In [7]:

```
df = spark.sql(  
    """  
    SELECT Image, ImageLoaded, Description, ProcessGuid  
    FROM mordorTable  
    WHERE Channel = "Microsoft-Windows-Sysmon/Operational"  
        AND EventID = 7  
        AND LOWER(ImageLoaded) IN (  
            'c:\\\\windows\\\\system32\\\\wbem\\\\scrcons.exe',  
            'c:\\\\windows\\\\system32\\\\vbscript.dll',  
            'c:\\\\windows\\\\system32\\\\wbem\\\\wbemdisp.dll',  
            'c:\\\\windows\\\\system32\\\\wshom.ocx',  
            'c:\\\\windows\\\\system32\\\\scrrun.dll'  
        )  
    )  
df.show(10, False)
```

Output:

Image	ImageLoaded	Description	P
C:\Windows\System32\wbem\scrcons.exe	C:\Windows\System32\wbem\scrcons.exe	WMI Standard Event Consumer - scripting {c4e35f20-f8ea-5f4e-c504-000000000400}	
C:\Windows\System32\wbem\scrcons.exe	C:\Windows\System32\vbscript.dll	Microsoft ® VBScript {c4e35f20-f8ea-5f4e-c504-000000000400}	
C:\Windows\System32\wbem\scrcons.exe	C:\Windows\System32\wbem\wbemdisp.dll	WMI Scripting {c4e35f20-f8ea-5f4e-c504-000000000400}	
C:\Windows\System32\wbem\scrcons.exe	C:\Windows\System32\wshom.ocx	Windows Script Host Runtime Library {c4e35f20-f8ea-5f4e-c504-000000000400}	
C:\Windows\System32\wbem\scrcons.exe	C:\Windows\System32\scrrun.dll	Microsoft ® Script Runtime {c4e35f20-f8ea-5f4e-c504-000000000400}	



Puedes Generar Impacto Siendo tu Mismo!

"Don't worry about being the next
me. Be the first you"

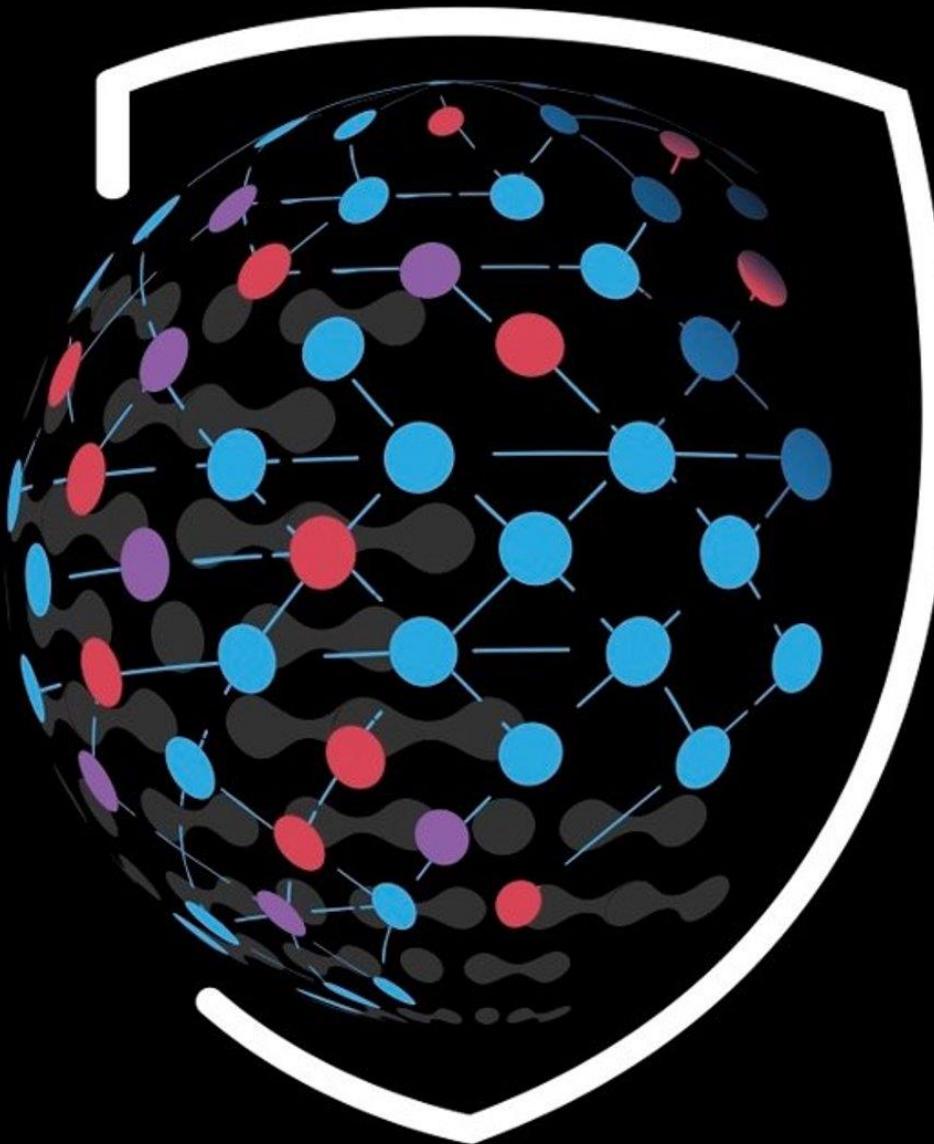
Dwayne 'The Rock' Johnson



Qué tal si
ponemos
todo junto?

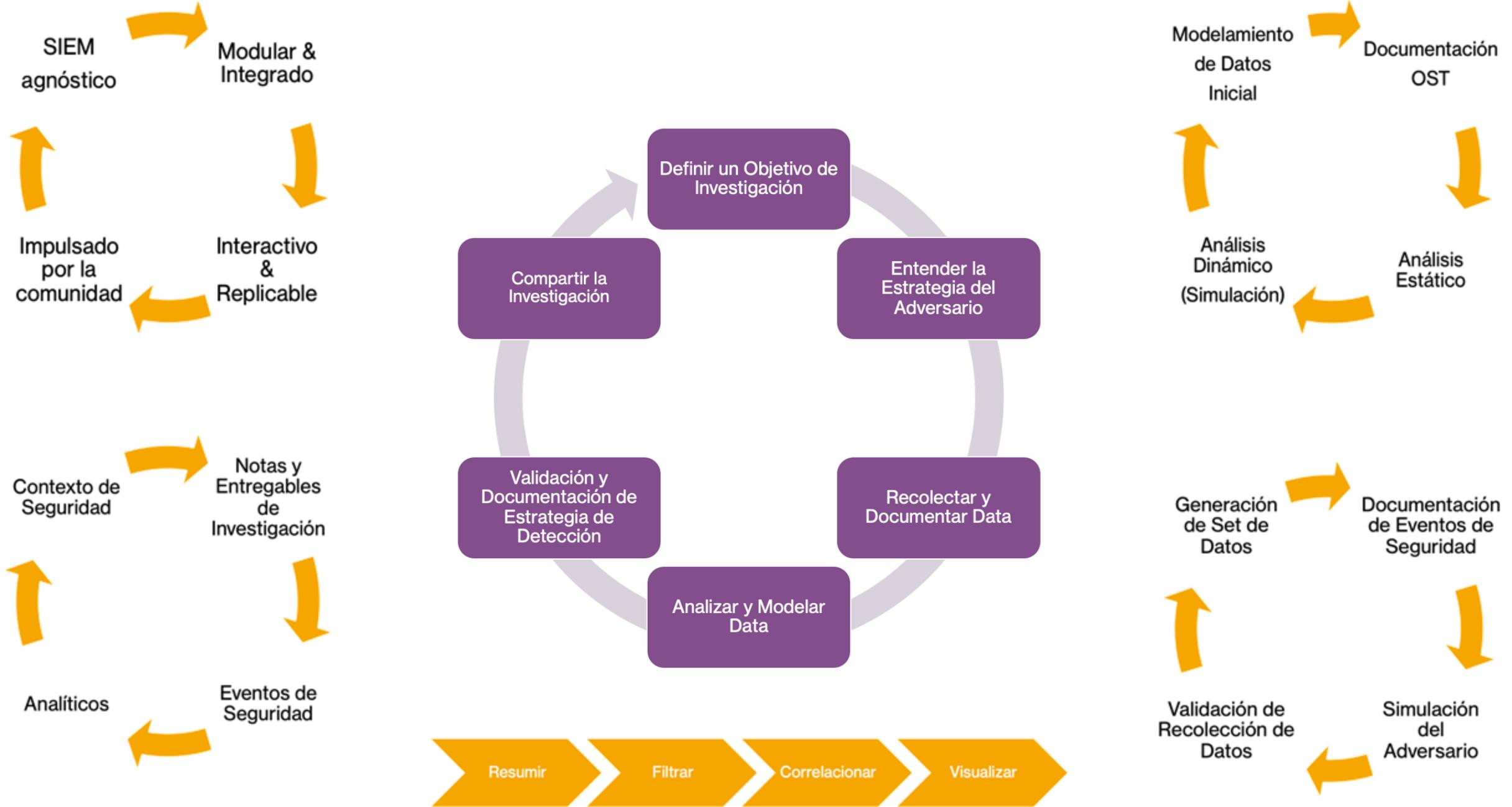
El poder de la comunidad!!

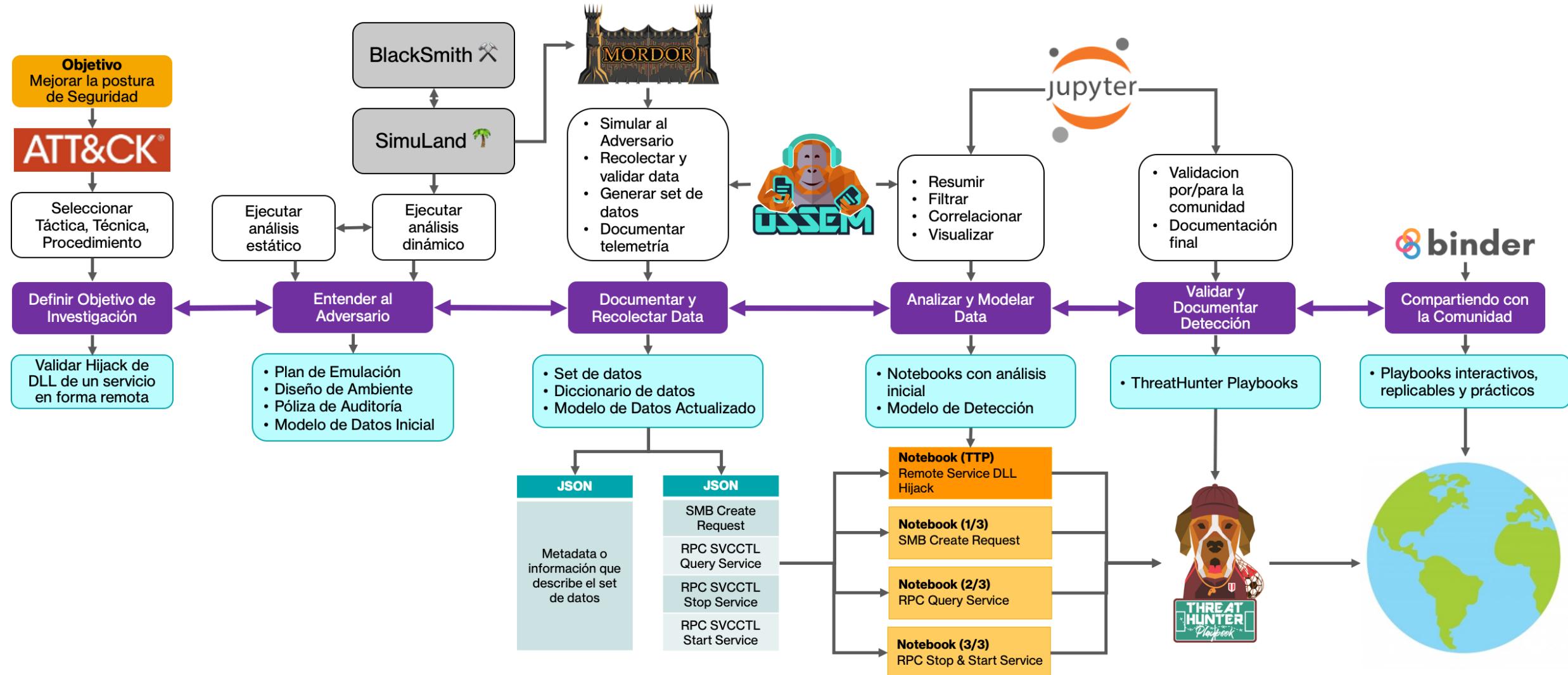




OPEN THREAT RESEARCH

EMPOWERING THE INFOSEC COMMUNITY





<https://github.com/OTRF>

The screenshot displays the GitHub repository page for "Open Threat Research Forge" (OTRF). The page features a dark-themed header with the repository name and a shield logo. Below the header, there are navigation links for "Repositories 18", "Packages", "People 8", "Teams 2", "Projects", and "Settings". The main content area is titled "Pinned repositories" and contains six repository cards:

- ThreatHunter-Playbook**: A Threat hunter's playbook to aid the development of techniques and hypothesis for hunting campaigns. (Python, 2.3k stars, 520 forks)
- mordor**: Re-play Adversarial Techniques. (Python, 717 stars, 95 forks)
- OSSEM**: Open Source Security Events Metadata (OSSEM). (Python, 675 stars, 129 forks)
- Blacksmith**: Building environments to replicate small networks and deploy applications. (PowerShell, 96 stars, 19 forks)
- Azure-Sentinel2Go**: Azure Sentinel2Go is an open source project developed to expedite the deployment of an Azure Sentinel lab. (Shell, 37 stars, 15 forks)
- infosec-jupyter-book**: The Infosec Community Definitive Guide to Jupyter Notebooks. (Dockerfile, 16 stars, 2 forks)

On the right side of the pinned repositories section, there is a link to "Customize pinned repositories".

OTR Discord Disponible



bitly.com/OTRDiscord

- Acepta el CAPTCHA
- Lee y acepta el código de conducta de nuestro discord
- Empezemos a colaborar y a compartir con la comunidad!

Partnership con el BlueSpace

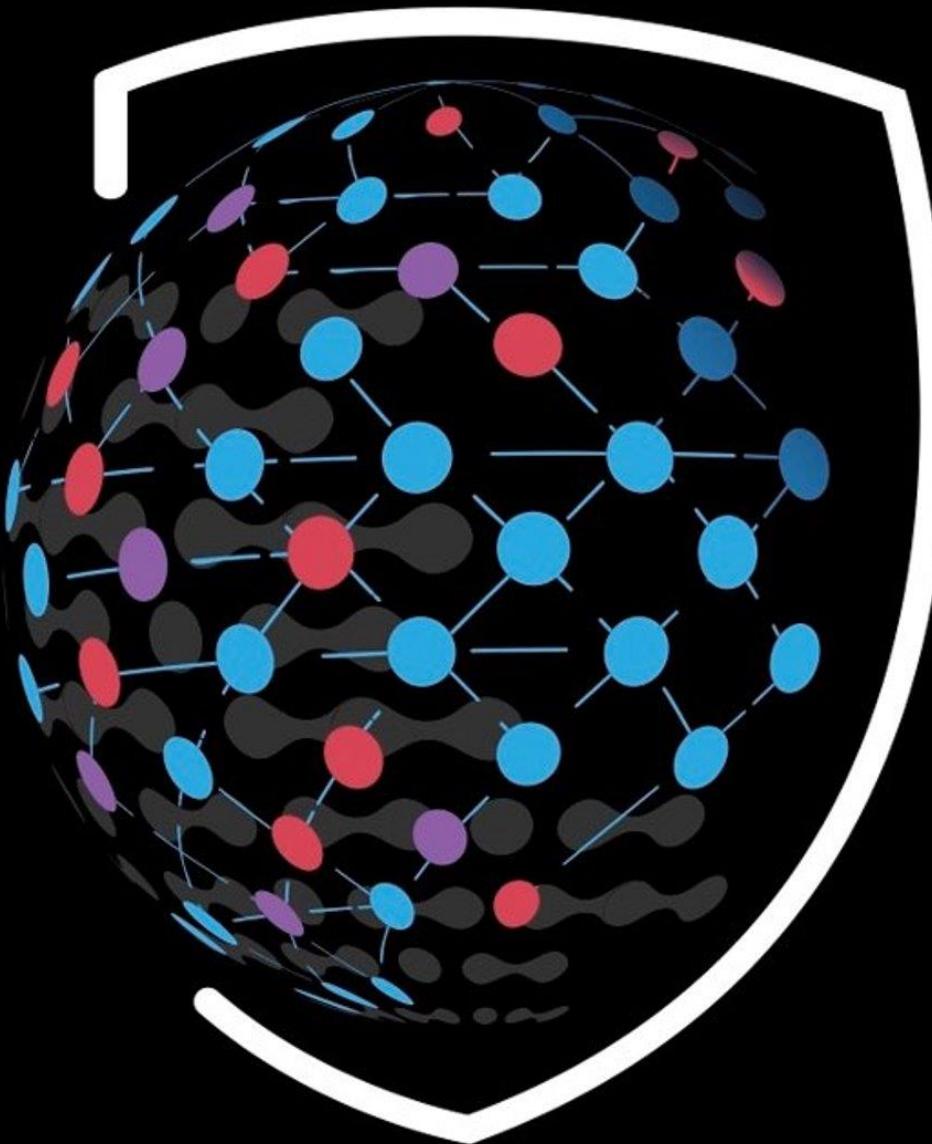


BLUESPACE

¿Quiénes somos? Ekoparty OTR Novedades Entradas Eko

BlueSpace y
Open Threat Research
partnership

<https://www.bluespacesec.org/otr>



OPEN THREAT RESEARCH

EMPOWERING THE INFOSEC COMMUNITY