black hat®
ARSENAL

AUGUST 7-8, 2024
MANDALAY BAY/LAS VEGAS

Kestrel 2: Hunt For Threats Across Security Data Lakes

Xiaokui Shu, Paul Coccoli, Edward Landis

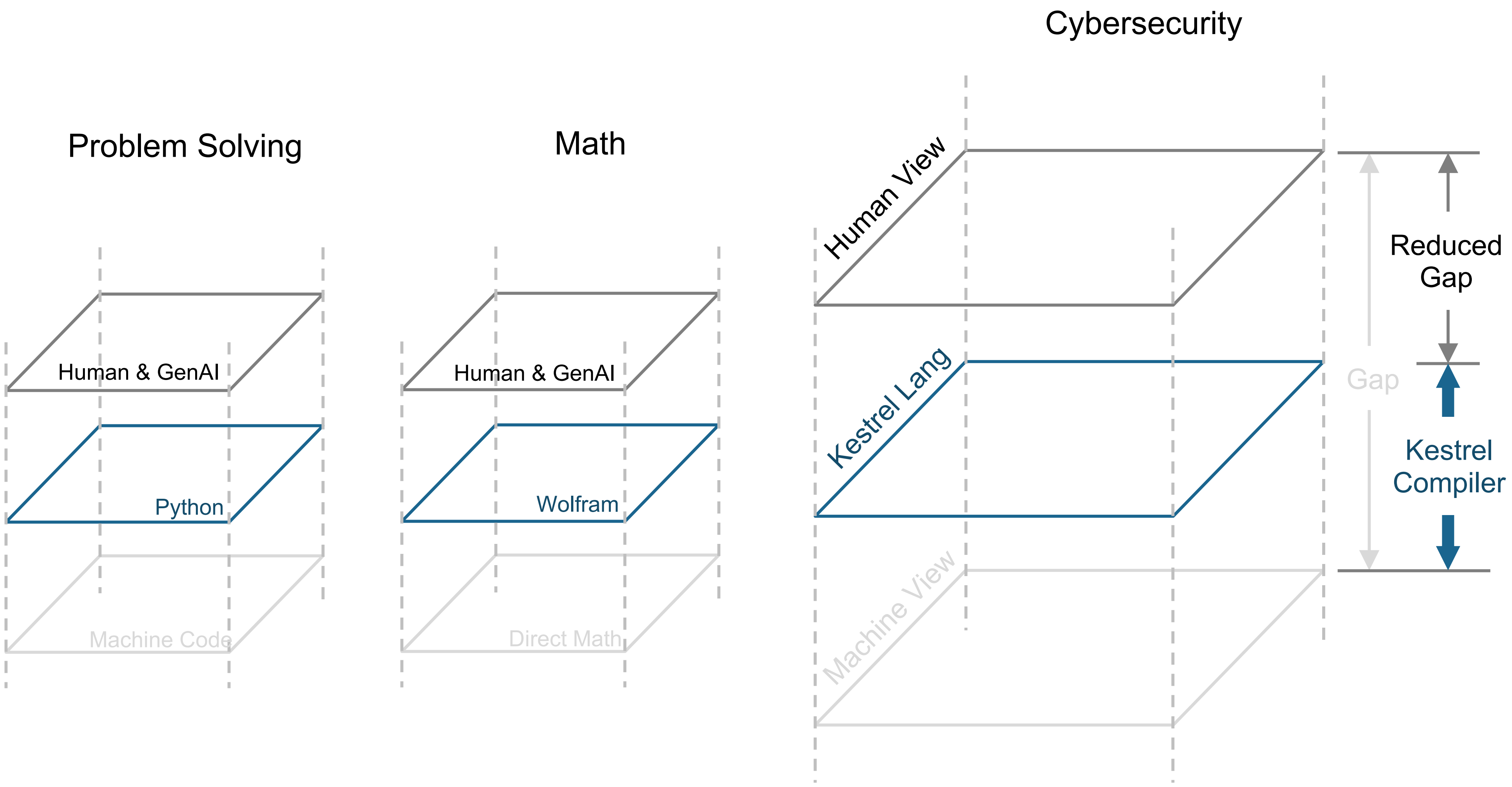
What to Learn in This Lab?

1. Kestrel
2. GoldenSAML

https://github.com/opencybersecurityalliance/kestrel-lang

OPEN
CYBERSECURITY
ALLIANCE

Kestrel



Problem Solving

Human & GenAI

Python

Machine Code

Math

Human & GenAI

Wolfram

Direct Math

Cybersecurity

Human View

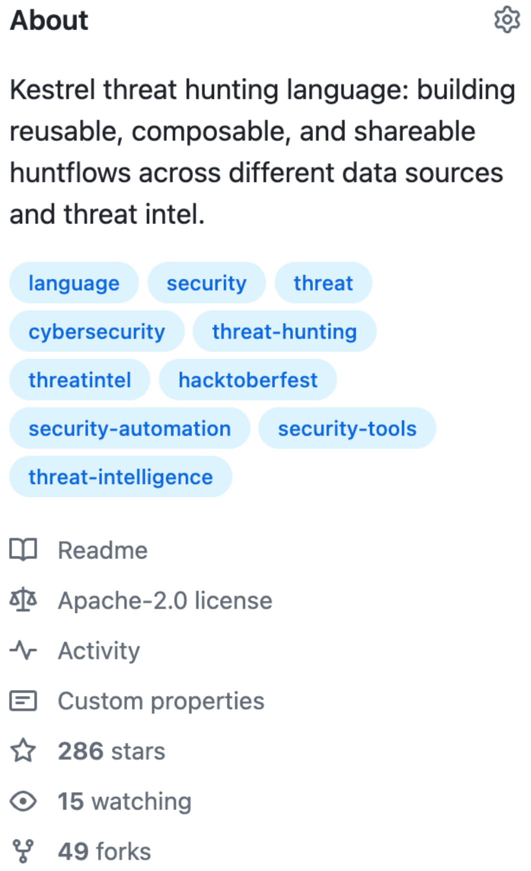
Kestrel Lang

Machine View

Reduced
Gap

Gap

Kestrel
Compiler



About

Kestrel threat hunting language: building reusable, composable, and shareable huntflows across different data sources and threat intel.

language security threat cybersecurity threat-hunting threatintel hacktoberfest security-automation security-tools threat-intelligence

Readme

Apache-2.0 license

Activity

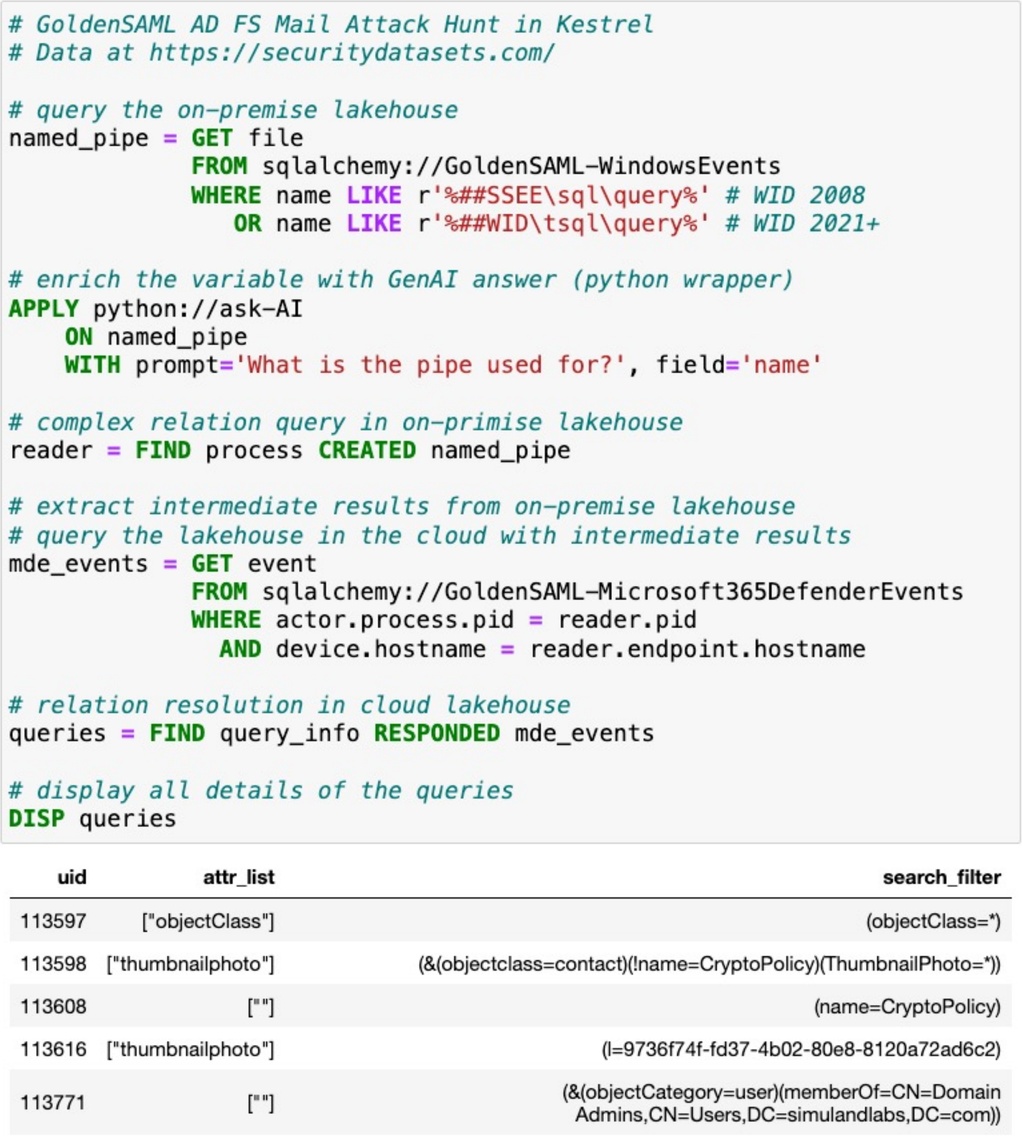
Custom properties

286 stars

15 watching

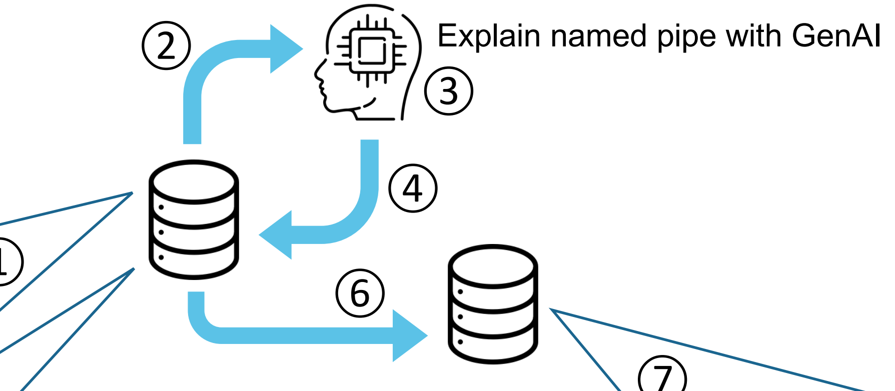
49 forks

Kestrel Under-The-Hood



```
# GoldenSAML AD FS Mail Attack Hunt in Kestrel
# Data at https://securitydatasets.com/

# query the on-premise lakehouse
named_pipe = GET file
             FROM sqlalchemy://GoldenSAML-WindowsEvents
             WHERE name LIKE r'%##SSEE\sql\query%' # WID 2008
                OR name LIKE r'%##WID\tsql\query%' # WID 2021+

# enrich the variable with GenAI answer (python wrapper)
APPLY python://ask-AI
    ON named_pipe
    WITH prompt='What is the pipe used for?', field='name'

# complex relation query in on-primise lakehouse
reader = FIND process CREATED named_pipe

# extract intermediate results from on-premise lakehouse
# query the lakehouse in the cloud with intermediate results
mde_events = GET event
             FROM sqlalchemy://GoldenSAML-Microsoft365DefenderEvents
             WHERE actor.process.pid = reader.pid
               AND device.hostname = reader.endpoint.hostname

# relation resolution in cloud lakehouse
queries = FIND query_info RESPONDED mde_events

# display all details of the queries
DISP queries
```

uid	attr_list	search_filter
113597	["objectClass"]	(objectClass=*)
113598	["thumbnailphoto"]	(&(objectclass=contact)(!name=CryptoPolicy)(ThumbnailPhoto=*))
113608	[""]	(name=CryptoPolicy)
113616	["thumbnailphoto"]	(l=9736f74f-fd37-4b02-80e8-8120a72ad6c2)
113771	[""]	(&(objectCategory=user)(memberOf=CN=Domain Admins,CN=Users,DC=simulandlabs,DC=com))



①

```
WITH named_pipe AS
  (SELECT DISTINCT "Computer" AS "endpoint.hostname",
                   "_ResourceId" AS "endpoint.uid",
                   "PipeName" AS name,
                   "EventID" AS type_id
   FROM windows
   WHERE "PipeName" LIKE '%##SSEE\sql\query%'
     OR "PipeName" LIKE '%##WID\tsql\query%')
SELECT DISTINCT *
FROM named_pipe
```

⑤

```
WITH reader AS
  (SELECT DISTINCT "Computer" AS "endpoint.hostname",
                   "_ResourceId" AS "endpoint.uid",
                   "Image" AS name,
                   "Image" AS "file.name",
                   "Image" AS "file.path",
                   "Image" AS "file.parent_folder",
                   "ProcessId_dynamic" AS pid,
                   "ProcessGuid" AS UID
   FROM windows
   WHERE ("PipeName",
          "_ResourceId") IN
       (SELECT DISTINCT *
        FROM "3e472784179c437eba728915c21332e0"))
SELECT DISTINCT pid
FROM reader
```

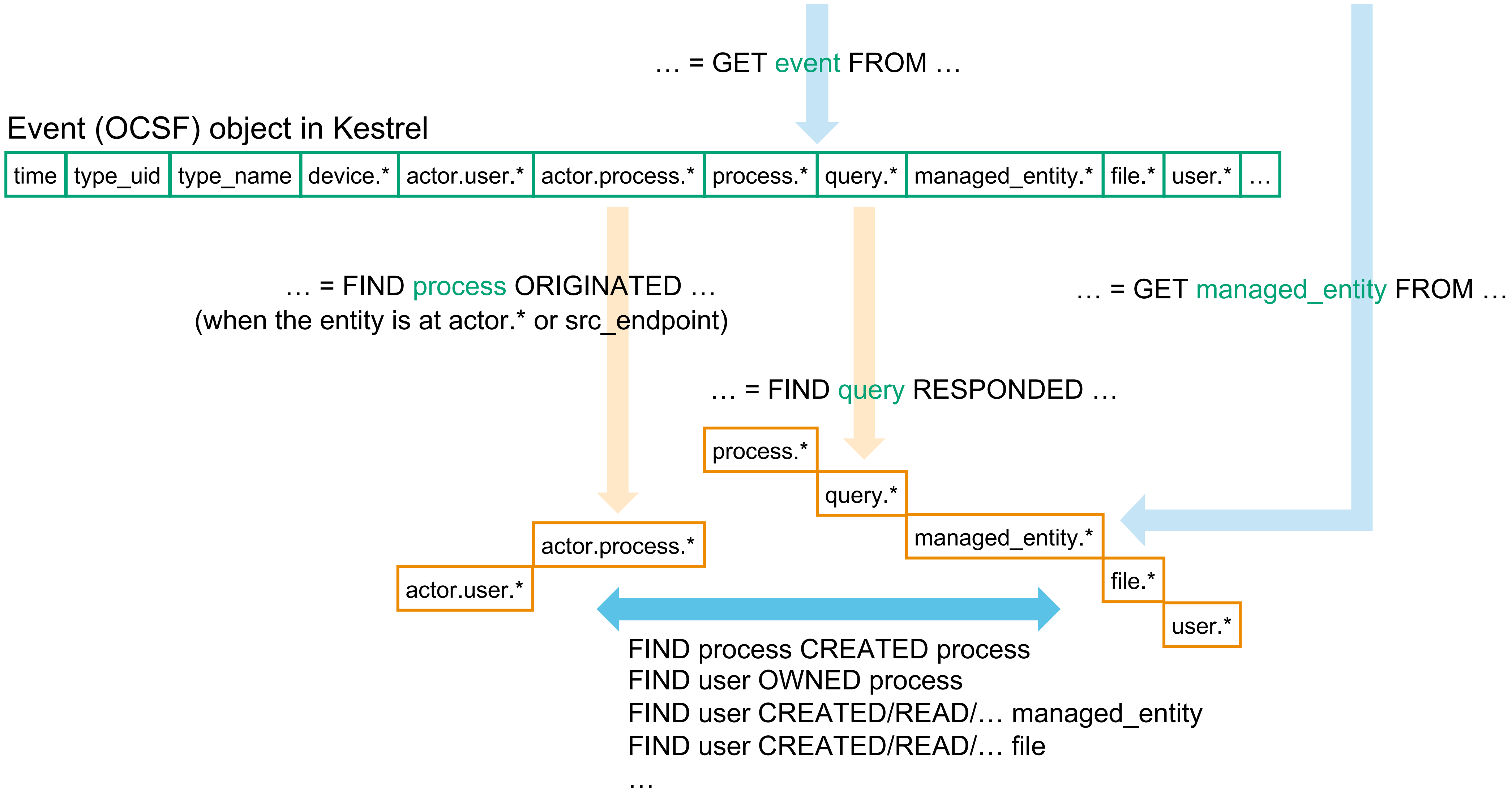
⑦

```
WITH mde_events AS
  (SELECT DISTINCT "Timestamp" AS TIME,
                   "DeviceName" AS "device.hostname",
                   "DestinationPort" AS "dst_endpoint.port",
                   "AccountDomain" AS "user.domain",
                   "AccountName" AS "user.name",
                   "AccountSid" AS "user.uid",
                   "UserAgent" AS "http_request.user_agent",
                   "ReportId" AS "query_info.uid",
                   "ReportId" AS "entity.uid",
                   "AdditionalFields.AttributeList" AS "query_info
                   "AdditionalFields.SearchFilter" AS "query_info.
                   "ActivityObjects" AS "entity.data",
                   "Permissions" AS PRIVILEGES,
                   "ActionType" AS type_uid,
                   "ActionType" AS type_name,
                   "RawEventData.ResultStatus" AS status_id
   FROM msdefender
   WHERE "InitiatingProcessId" IN
       (SELECT DISTINCT *
        FROM "273344ee8bfc4ef08e035d89cb0538d6")
     AND "DeviceName" IN
       (SELECT DISTINCT *
        FROM "4058b03845074fd2bda04e2545f520d0")),
     queries AS
  (SELECT DISTINCT "query_info.uid" AS UID,
                   "query_info.attr_list" AS attr_list,
                   "query_info.search_filter" AS search_filter
   FROM mde_events
   WHERE 1 = 1)
SELECT DISTINCT *
FROM queries
```

Kestrel Language Cheat Sheet

Raw line of log (from datasource):

```
{"TimeGenerated":"2021-08-02T13:05:32.77Z","Source":"Microsoft-Windows-Sysmon","EventLog":"Microsoft-Windows-Sysmon/Operational","Computer":"ADFS01.simulandlabs.com",…
```



… = GET event FROM …

Event (OCSF) object in Kestrel

time	type_uid	type_name	device.*	actor.user.*	actor.process.*	process.*	query.*	managed_entity.*	file.*	user.*	…

… = FIND process ORIGINATED …
(when the entity is at actor.* or src_endpoint)

… = GET managed_entity FROM …

… = FIND query RESPONDED …

FIND process CREATED process
FIND user OWNED process
FIND user CREATED/READ/… managed_entity
FIND user CREATED/READ/… file
…

Commands Supported (in Kestrel 2 Beta)

- variable = NEW …
- variable = GET …
- variable = FIND …
- DISP variable …
- INFO variable
- APPLY … ON variable …
- EXPLAIN variable

Full Documentation:
https://kestrel.readthedocs.io

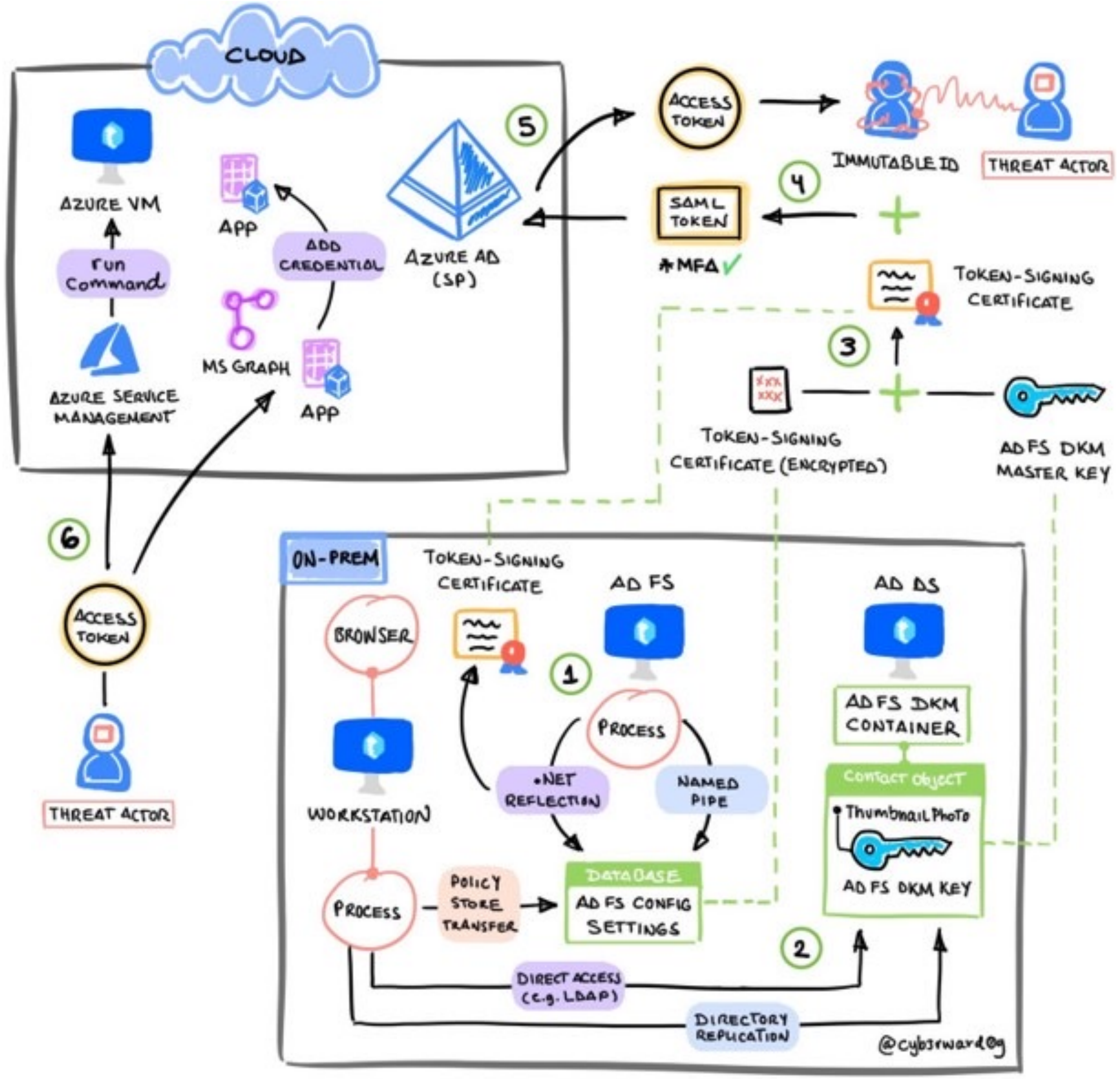
GoldenSAML

Background & Overview

A Golden SAML attack targets an organization's authentication process by exploiting the Security Assertion Markup Language (SAML) used by the organization's identity provider (IdP) to issue tokens for cloud applications & services.

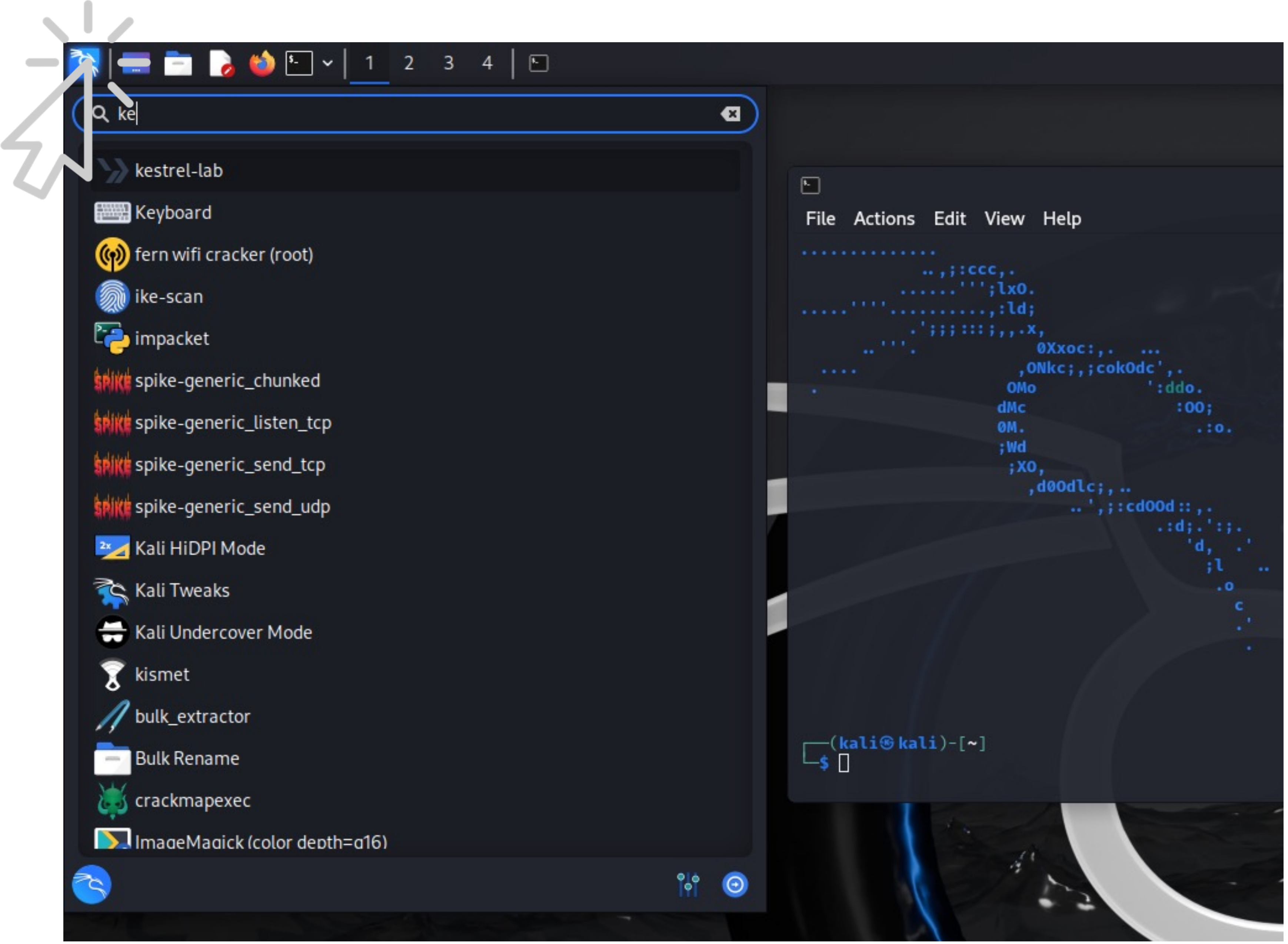
Attacker's Goals

- Gain access to the IdP
- Steal the private key used to sign SAML tokens
- Forge their own SAML Token
- Access the cloud resource by impersonating a user



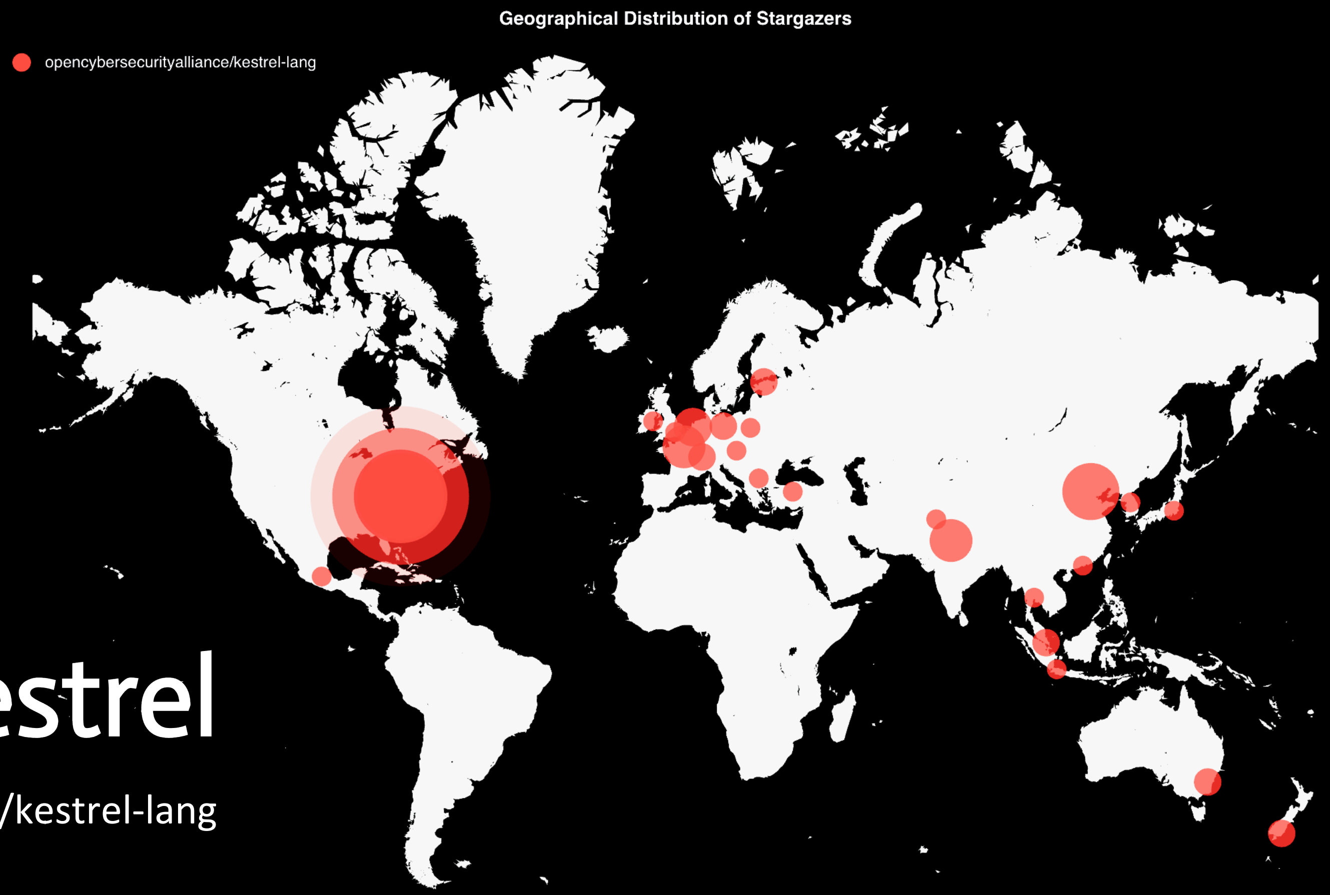
https://simulandlabs.com/labs/GoldenSAML/README.html

Let's Hunt...



```
┌──(kali㉿kali)-[~]
└─$ 
```

Happy Hunting!


Geographical Distribution of Stargazers
● opencybersecurityalliance/kestrel-lang

Kestrel

https://github.com/opencybersecurityalliance/kestrel-lang