

SIGMA UNLEASHED: A REALISTIC IMPLEMENTATION



IMPORTANT NOTICE

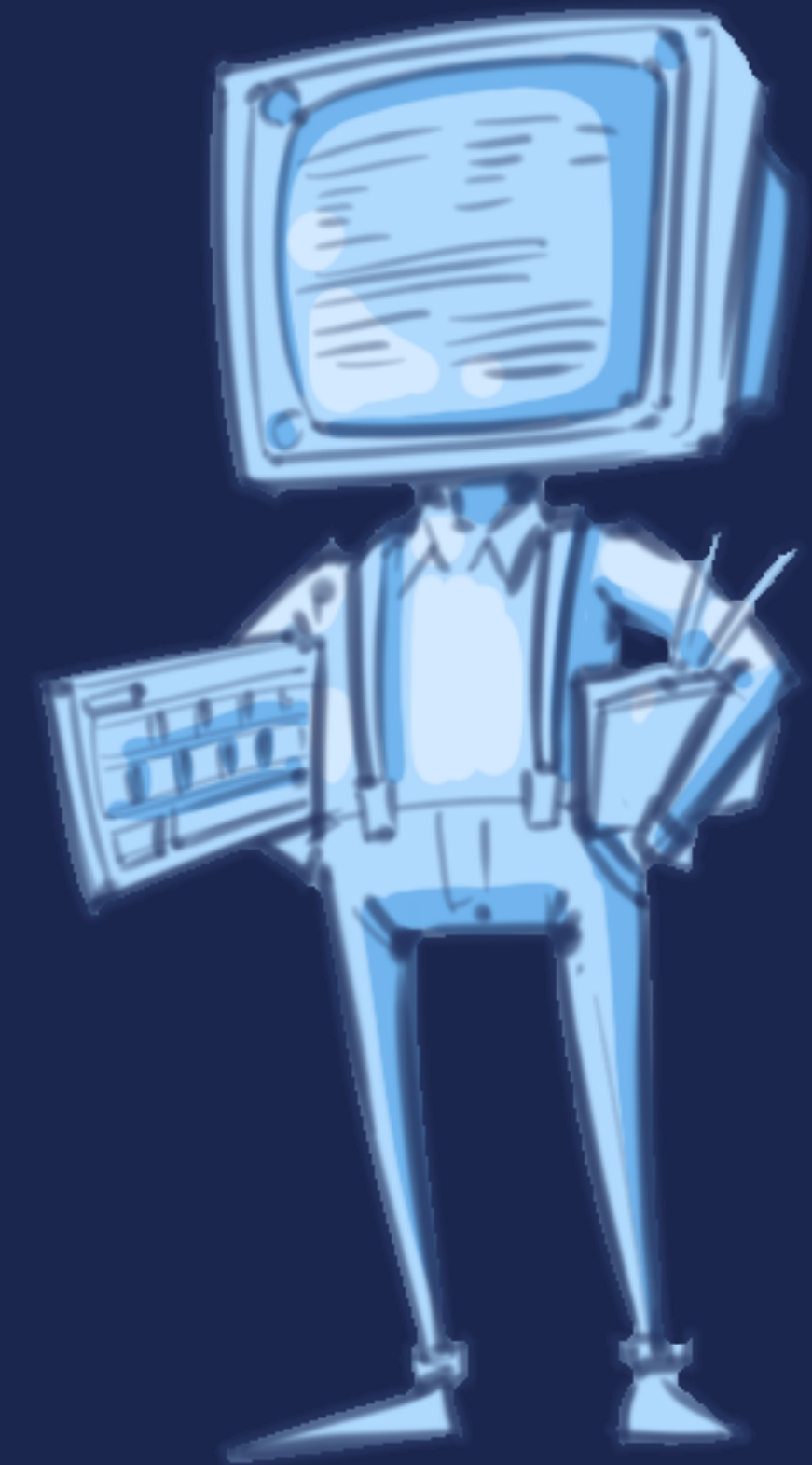
THIS PRESENTATION IS **TLP:CLEAR**

SUBJECT TO STANDARD COPYRIGHT RULES, YOU MAY
SHARE THIS PRESENTATION WITHOUT RESTRICTION

© CERT-EU, CERT FOR THE INSTITUTIONS, BODIES, AND
AGENCIES

AGENDA

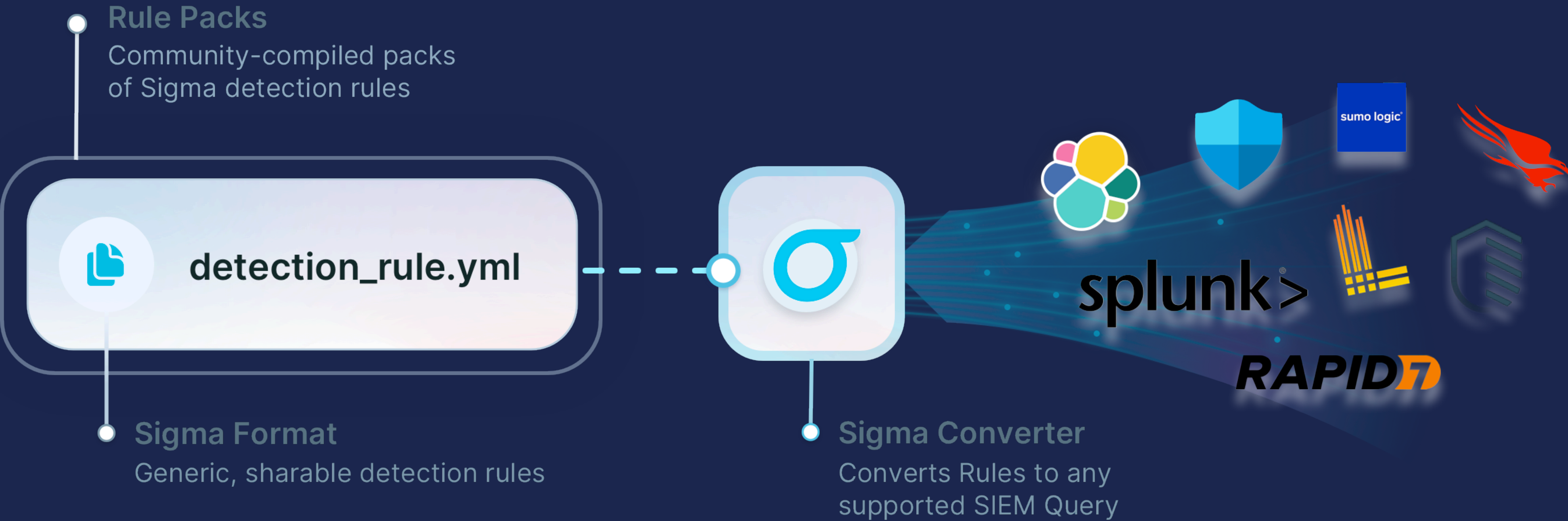
- ▶ What is Sigma?
- ▶ Our context and Detection-as-Code workflow
- ▶ The tool: droid
- ▶ Conclusion
 - ▶ ... key takeaways
 - ▶ ... references





SIGMA

SIGMA



Source: sigmahq.io

HOW DOES IT LOOK LIKE?

- ▶ **YAML** format
- ▶ Open and **generic**
- ▶ **Easy** to read
- ▶ Encourages **sharing**

```
aws_root_account_usage.yml

title: AWS Root Credentials
description: Detects AWS root account usage
logsource:
  product: aws
  service: cloudtrail
detection:
  selection:
    userIdentity.type: Root
  filter:
    eventType: AwsServiceEvent
  condition: selection and not filter
falsepositives:
  - AWS Tasks That Require Root User Credentials
level: medium
```

```
$ sigma convert -t splunk -p config.yml ...
```



```
sourcetype="aws:cloudtrail" userIdentity.type="Root"
NOT eventType="AwsServiceEvent"
```

... or **any supported SIEM**.

Source: sigmahq.io

POWERED BY PYSIGMA SINCE 2023

- ▶ pySigma parses and **converts** Sigma rules into queries
- ▶ The **backends** support the conversion
- ▶ The **pipelines** transforms the rule based on a set of conditions



Example Provided: Splunk Queries using the pysigma-backend-splunk backend

Source: sigmahq.io

SIGMA COMES WITH A CONVERTER

sigconverter.io
sigma rule converter

Backend:

splunk

Format:

default

Pipeline:

select pipelines...

CLI:

```
sigma convert --without-pipeline -t splunk -f default rule.yml
```



rule.yml

pipeline.yml

```
title: Suspicious SYSTEM User Process Creation
id: 2617e7ed-adb7-40ba-b0f3-8f9945fe6c09
status: test
description: Detects a suspicious process creation as SYSTEM user (suspicious)
references:
  - Internal Research
  - https://tools.thehacker.recipes/mimikatz/modules
author: Florian Roth (rule), David ANDRE (additional keywords)
date: 2021/12/20
modified: 2022/04/27
logsource:
  category: process_creation
  product: windows
```

query

```
IntegrityLevel="System" User IN ("*AUTHORI*", "*AUTORI*") Image IN ("*\
\calc.exe", "*\wscript.exe", "*\cscript.exe", "*\hh.exe", "*\
\mshta.exe", "*\forfiles.exe", "*\ping.exe") OR CommandLine IN ("* -NoP
*", "* -W Hidden *", "* -decode *", "* /decode *", "* /urlcache *", "* -
urlcache *", "* -e* JAB*", "* -e* SUVYI*", "* -e* SQBFAFgA*", "* -e*
aWV4I*", "* -e* IAB*", "* -e* PAA*", "* -e* aQBlAHgA*", "*vssadmin delete
shadows*", "*reg SAVE HKLM*", "* -ma *", "*Microsoft\\Windows\
\CurrentVersion\\Run*", "*downloadstring*", "*downloadfile*", "* /
ticket:*", "*dpapi:*", "*event::clear*", "*event::drop*", "*id::modify*",
"*kerberos:*", "*lsadump:*", "*misc:*", "*privilege:*", "*rpc:*",
"*sekurlsa:*", "*sid:*", "*token:*", "*vault::cred*", "*vault::list*",
"* p::d *", "*;iex*", "*MiniDump*", "*net user *")
```


SIGMA PROCESSING PIPELINE 101

```
name: Splunk Windows Process Access
priority: 100

# Author: Mathieu LE CLEACH
# Purpose of this pipeline:
# Process the windows/process_access rules for Splunk

transformations:
  - id: index_condition
    type: add_condition
    conditions:
      index: windows_splunk_access
      splunk_server: "*prod.planet-express.local"
    rule_conditions:
      - type: logsource
        category: process_access
        product: windows

postprocessing:
  - type: template
    template: |+
      {{ query }} | table _time,host,user,SourceImage,TargetImage,CallTrace,GrantedAccess
```

UNLEASHING SIGMA IS NOT EASY

- ▶ Works out of the box for most cases
- ▶ ... but often criticised





OUR CONTEXT

NAVIGATING CHALLENGES

- ▶ We serve **90+** Union Entities
- ▶ We do security log **monitoring**
- ▶ We operate like an **MSSP**
 - ▶ ... Cloud and On-Premises
 - ▶ ... **Multiple** SIEM/EDR

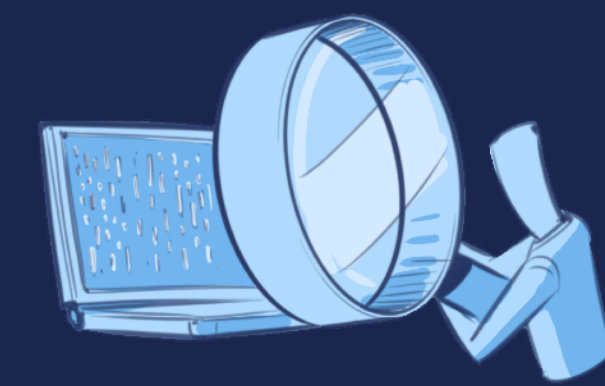


DETECTION-AS-CODE AT RESCUE

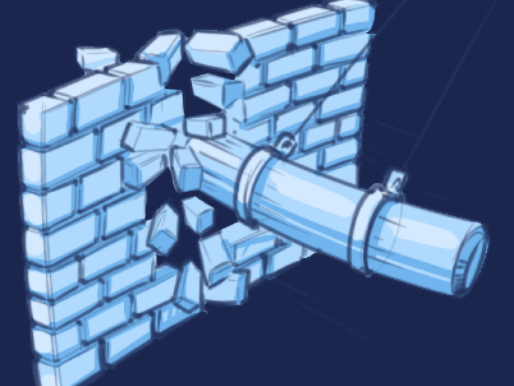
CONTENT VERSIONING

MODULARITY AND QA TESTING

VENDOR AGNOSTIC



Threat Hunting



Post-breach



Threat Intelligence



Detection objectives




splunk>



DETECTION-AS-CODE BENEFITS

- ▶ Improved **collaboration**
- ▶ Four-eyes policy reduce false positives, hence **alert fatigue**
- ▶ Ensure a continuous and **iterative** process

Release: Adding more rules to detect APT0 (Evil Bobcat)

 Merged Mathieu Le Cleach requested to merge `develop` into `main` 1 month ago

Overview 0 Commits 5 Pipelines 4 Changes 11

Description

Following the investigation of the last APT0 engagement, the [investigation](#) revealed several detection opportunities to detect an early compromise by Evil Bobcat.

 Merge request pipeline #196346 passed
Merge request pipeline passed for `9ffdf486` 1 month ago




8✓ Approved by 

 Merged by  [Mathieu Le Cleach](#) 1 month ago

[Revert](#) [Cherry-pick](#)

Merge details

- Changes merged into `main` with [fee9c954](#).
- Did not delete the source branch.

 Pipeline #196353 passed
Pipeline passed for `fee9c954` on `main` 1 month ago



Deployed to [production](#) 1 month ago





DROID

DROID: A PYSIGMA WRAPPER



- ▶ **Validate** the rules (syntax)
- ▶ **Convert** the Sigma rules
- ▶ **Search** and report for findings
- ▶ Check the **integrity** of the rules
- ▶ **Export** (deploy) Sigma rules to SIEM/EDR



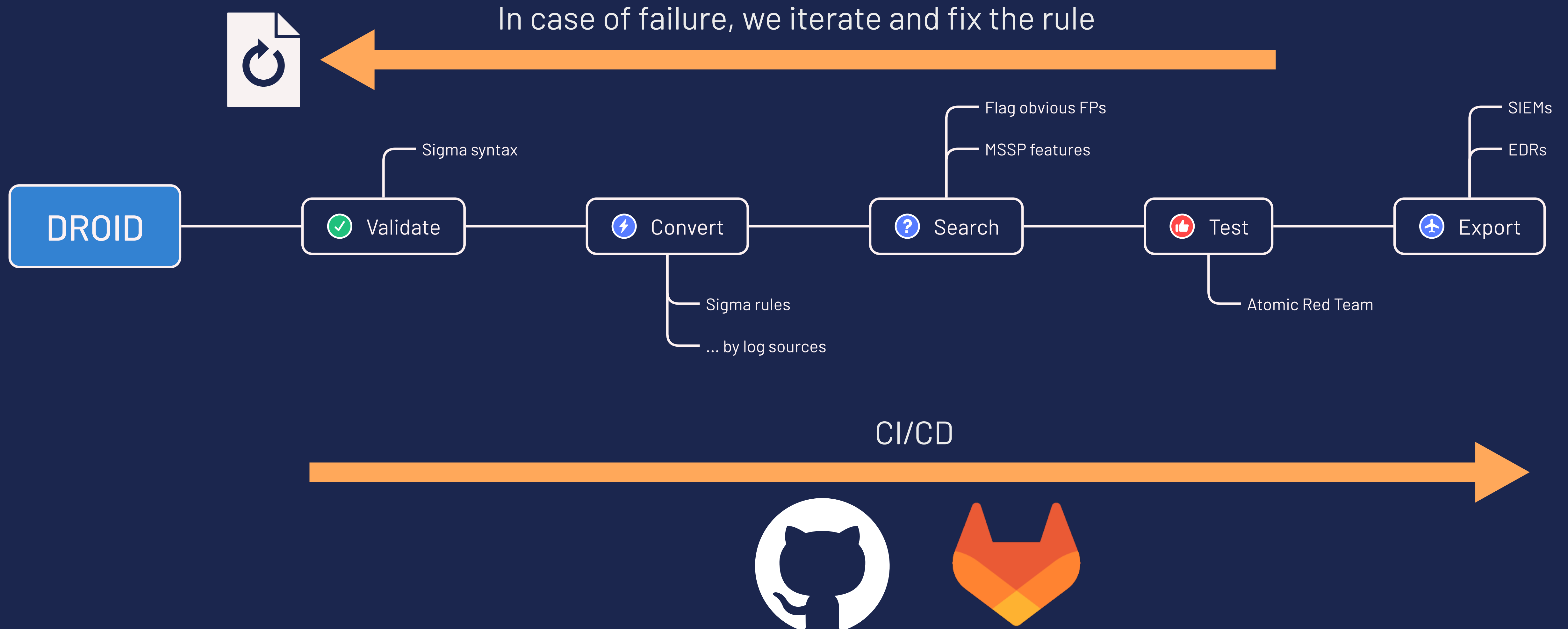
- ▶ Supported format: **Sigma** and plain SIEM/EDR **native** query

A CONFIG TO RULE THEM ALL

- ▶ TOML configuration file
- ▶ Highly flexible
- ▶ Sigma conversion is done by platform and log sources

```
1  # This is a TOML document config
2
3  title = "droid configuration file"
4
5  ∨ [base]
6
7  raw_rules_directory = "rules/raw"
8
9  ▶ sigma_validation_config = "validation/validate.yml"
10
11 ∨ [platforms]
12
13 > [platforms.azure] ...
64 ∨ > [platforms.microsoft_defender] ...
97 > [platforms.splunk] ...
```

DROID WORKFLOW



RULE SYNTAX VALIDATION

 Validate

- ▶ Leverage **pySigma** validators
- ▶ Validate Sigma rules in a **configurable** way
- ▶ Well-suited for integration within a **CI/CD** pipeline

```
validation > ! validate.yml > ...
1   validators:
2       - all
3       - -tlpv1_tag
4       - -escaped_wildcard
5   exclusions:
6       5c84856b-55a5-45f1-826f-13f37250cf4e:
7       - number_as_string
```

```
$ droid --config-file config.toml --validate -r ./rules/sigma/
```

```
2024-05-21 12:46:54,064:INFO:droid:Validation mode was selected - path selected: rules/sigma/
Invalid MITRE ATT&CK tagging - severity MEDIUM at:
    rules/sigma/sigmahq/windows/process_access/proc_access_win_lsass_werfault.yml
2024-05-21 12:46:54,112:ERROR:droid:Validation issues found
```

RULE CONVERSION BASED ON THE LOG SOURCES 1/2



splunk®>

```
[platforms.microsoft_defender.pipelines.windows_process_creation]

pipelines = ["pipelines/mde_process_creation.yml", "microsoft_365_defender"]
product = "windows"
category = "process_creation"

[platforms.microsoft_defender.pipelines.windows_process_creation_thread]

pipelines = ["pipelines/mde_process_creation_thread.yml", "microsoft_365_defender"]
product = "windows"
category = "process_creation"
service = "create_remote_thread"

[platforms.microsoft_defender.pipelines.windows_fileevent]

pipelines = ["microsoft_365_defender"]
product = "windows"
category = "file_event"
```

```
[platforms.splunk.pipelines.windows_process_access]

pipelines = ["pipelines/splunk_process_access.yml", "splunk_windows"]
product = "windows"
category = "process_access"

[platforms.splunk.pipelines.windows_image_load]

pipelines = ["pipelines/splunk_windows_image_load.yml"]
product = "windows"
category = "image_load"

[platforms.splunk.pipelines.windows_process_creation]

pipelines = ["splunk_windows", "pipelines/splunk_process_creation.yml"]
product = "windows"
category = "process_creation"
format = "data_model"
```


RULE CONVERSION BASED ON THE LOG SOURCES 2/2

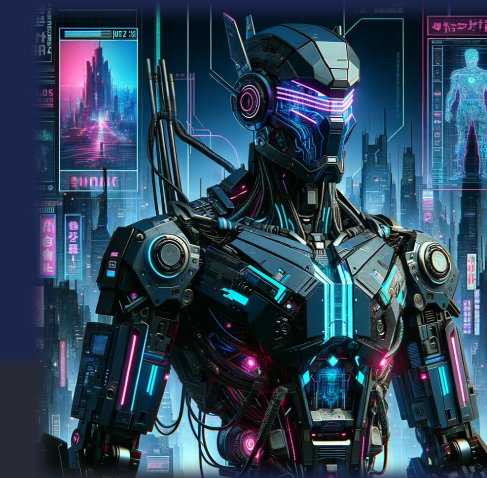
 Convert

```
author: Nasreddine Bencherchali (Nextron Systems)
date: 2022/06/27
modified: 2023/05/15
Add Tag
tags:
  - attack.defense_evasion
  - attack.t1055
logsource:
  category: process_creation
  product: windows
```

SIGMA
RULE

```
[platforms.splunk.pipelines.windows_process_creation]
```

```
pipelines = ["splunk_windows", "pipelines/splunk_process_creation.yml"]
product = "windows"
category = "process_creation"
format = "data_model"
```



ROCK, ROBOT ROCK

 Convert

```

✓ pipelines
  ! splunk_process_access.yml
✓ rules / sigma
  ✓ sigmahq / windows / process_access
    ⚙️ proc_access_win_lsass_werfault.yml
  .gitkeep
  > validation
  ⚙️ droid_config.toml

```



category: process_access
product: windows

```

name: Splunk Windows Process Access
priority: 100

# Author: Mathieu LE CLEACH
# Purpose of this pipeline:
# Process the windows/process_access rules for Splunk

transformations:
  - id: index_condition
    type: add_condition
    conditions:
      index: windows_splunk_access
      splunk_server: "*prod.planet-express.local"
    rule_conditions:
      - type: logsource
        category: process_access
        product: windows

postprocessing:
  - type: template
    template: |+
      {{ query }} | table _time,host,user,SourceImage,TargetImage,CallTrace,
      GrantedAccess

```

```
$ droid --config-file config.toml --convert --platform splunk -r ./rule.yml
```

```

index="windows_splunk_access" splunk_server="*prod.planet-express.local" SourceImage="*\\WerFault.exe"
TargetImage="*\\lsass.exe" GrantedAccess="0x1FFFFFF" | table _time,host,user,SourceImage,TargetImage,
CallTrace,GrantedAccess

```


SEARCH FOR ANY RESULT



- ▶ **Search** in your dataset and look for any result
- ▶ ... uncover potential **false positives**

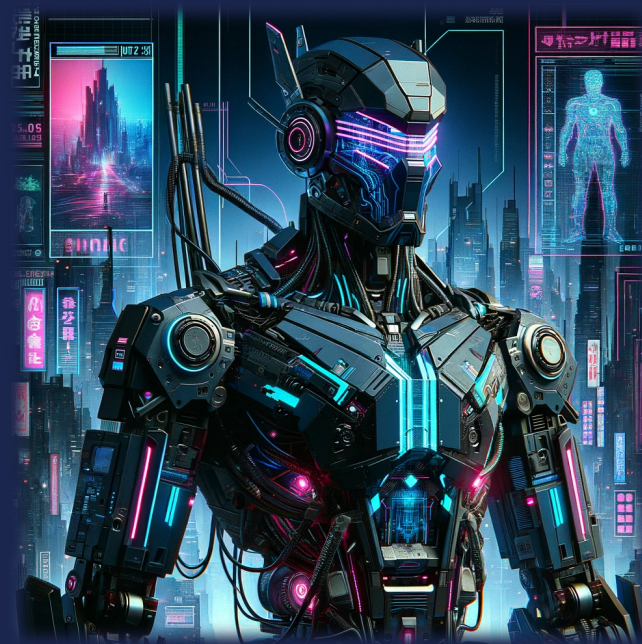


```
$ droid --config-file config.toml --search --platform splunk -r ./rule.yml
```

```
2024-05-23 22:30:37,921:INFO:droid.platforms.splunk.SplunkPlatform:Searching for rule rules/sigma/sigmahq/windows/process_creation/p
2024-05-23 22:31:29,714:INFO:droid.search:Successfully searched the rule rules/sigma/sigmahq/windows/process_creation/proc_creation_
2024-05-23 22:31:29,714:WARNING:droid.search:(Splunk) Match found for rules/sigma/sigmahq/windows/process_creation/proc_creation_win
_detection_rules/search?sid=1716496237.1165215
```

- ▶ MS Sentinel: **search** in multiple workspaces
 - ▶ (requires an Azure Lighthouse setup)





Executes Atomic tests remotely



Atomic Red Team™

- ▶ A Sigma rule is linked to a MITRE technique ID and a test GUID
- ▶ Test pass if the activity was detected in the SIEM/EDR



splunk®

Disclaimer: while we use the techniques and methodologies provided by Atomic Red Team™, our tool is an independent project and is not endorsed or sponsored by Red Canary.

- ▶ Highly flexible
- ▶ Export config reflects a general config
- ▶ Built for CI/CD

```
[platforms.splunk.savedsearch_parameters]

alert_type = "number of events"
app = "detection_rules_app"
sharing = "app"
alert_comparator = "greater than"
alert_threshold = 0
"alert.track" = 1
allow_skew = "67%"
"alert.suppress" = 1
"alert.digest_mode" = 0 # Per result (per row)
"alert.suppress.period" = "8h"
"alert.suppress.fields" = "customer_name,host"

# [platforms.splunk.savedsearch_parameters.suppress_fields_groups.group_name]
# Optional: define a suppress fields groups by logsource (.e.g. web)

[platforms.splunk.savedsearch_parameters.suppress_fields_groups.windows_image_load]

category = "image_load"
product = "windows"
"alert.suppress.fields" = "customer_name,host,ImageLoaded"

[platforms.splunk.action]

actions = "webhook"
"action.webhook.param.url" = "webhook"
```


RULE LEVEL EXPORT CONFIGURATION



- ▶ Apply **custom** saved search settings
- ▶ Disable rule
- ▶ Remove rule
- ▶ **Overwrite** actions
 - ▶ ... e.g. webhook URL

```
- all_trusts # Flag for /domain_trusts
- 'dclist:'
- 'dnsgetdc:'
- 'domain_trusts'
- 'dsgetdc:'
- 'parentdomain'
- 'trusted_domains'

condition: all of selection_*
falsepositives:
  - Legitimate administration use but user and host must be investigated
level: low
custom:
  cron_schedule: '* /5 * * * *'
  earliest_time: "-15m@m"
```

SIGMA
RULE

► Export your rule

```
$ droid --config-file config.toml --export --platform splunk -r ./rule.yml
```


```
2024-05-23 23:24:29,323:INFO:droid.platforms.splunk.SplunkPlatform:Saved search for rule rules/sigma/sigmahq/windows/  
2024-05-23 23:24:29,323:INFO:droid.export:Successfully exported the rule rules/sigma/sigmahq/windows/process_creation  
2024-05-23 23:24:29,323:INFO:droid:Successfully exported the rules
```

► Check its configuration

```
$ droid --config-file config.toml --integrity --platform splunk -r ./rule.yml
```

```
2024-05-23 23:25:35,512:INFO:droid.integrity:Successfully retrieved the rule rules/sigma/sigmahq/windows/process_  
2024-05-23 23:25:35,512:DEBUG:droid.integrity:detection in rule_content matches search in result  
2024-05-23 23:25:35,512:DEBUG:droid.integrity:description in rule_content matches description in result  
2024-05-23 23:25:35,513:INFO:droid.integrity:The rule is enabled as expected  
2024-05-23 23:25:35,513:INFO:droid:Integrity check successful
```


SearchAnalyticsDatasetsReportsAlertsDashboards

 **CERT-EU**
Detection rules

CobaltStrike Service Installations - System

Edit ▾

Detects known malicious service installs that appear in cases in which a Cobalt Strike beacon elevates privileges or lateral movement

Enabled: Yes.

App:


Permissions: Shared in App. Owned


Modified: Apr 26, 2024 6:14:59 PM


Alert Type: Scheduled. Cron Schedule.

Trigger Condition: .. Number of Results is > 0.


Actions: ▾ 2 Actions


 Add to Triggered Alerts

 Webhook

 [CERT-EU] [DEV] Potential Reconnaissance For Cached Credentials Via Cmdkey.EXE

High
Severity

 Custom
Content Source

 Enabled
Status

InfoInsights

ID

07f8bdc2-c9b3-472a-9817-5a670b872f53

Description

Detects usage of cmdkey to look for cached credentials on the system

Rule query

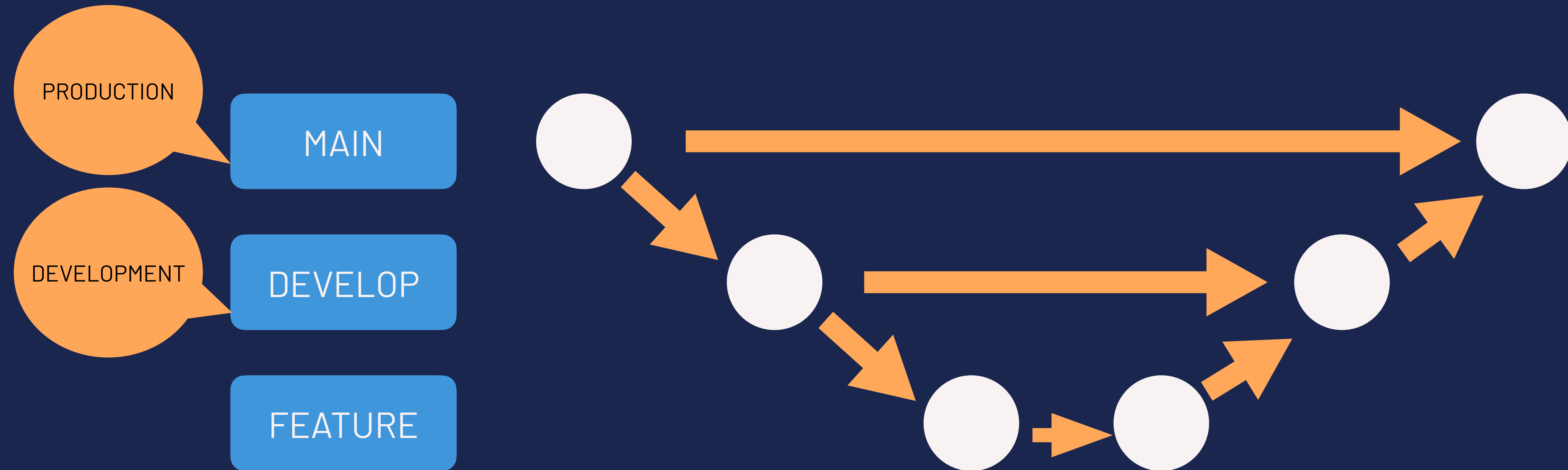
DeviceProcessEvents
| where (FolderPath endswith "\\cmdkey.exe" or Proce

TLP:CLEAR




USAGE

DETECTION CONTENT AND GIT FLOW



- ▶ Enhanced **testing** and **validation**
- ▶ Improved collaboration and reduce risk of **false positives**
- ▶ CI/CD integration in **production** and **development** env

HOW TO GET STARTED 1/2

- ▶ Get familiar with:
 - ▶ ... any DevOps toolset (mainly **git**)
 - ▶ ... **Sigma** concept and rule conversion
- ▶ Initiate your own repo by cloning **init-droid**
- ▶ <https://github.com/certeu/droid-init> 

```
└─ pipelines
   └─ ! splunk_process_access.yml
└─ rules / sigma
   └─ sigmahq / windows / process_access
      └─ 🌀 proc_access_win_lsass_werfault.yml
         └─ 📁 .gitkeep
            > validation
               └─ ⚙️ droid_config.toml
```

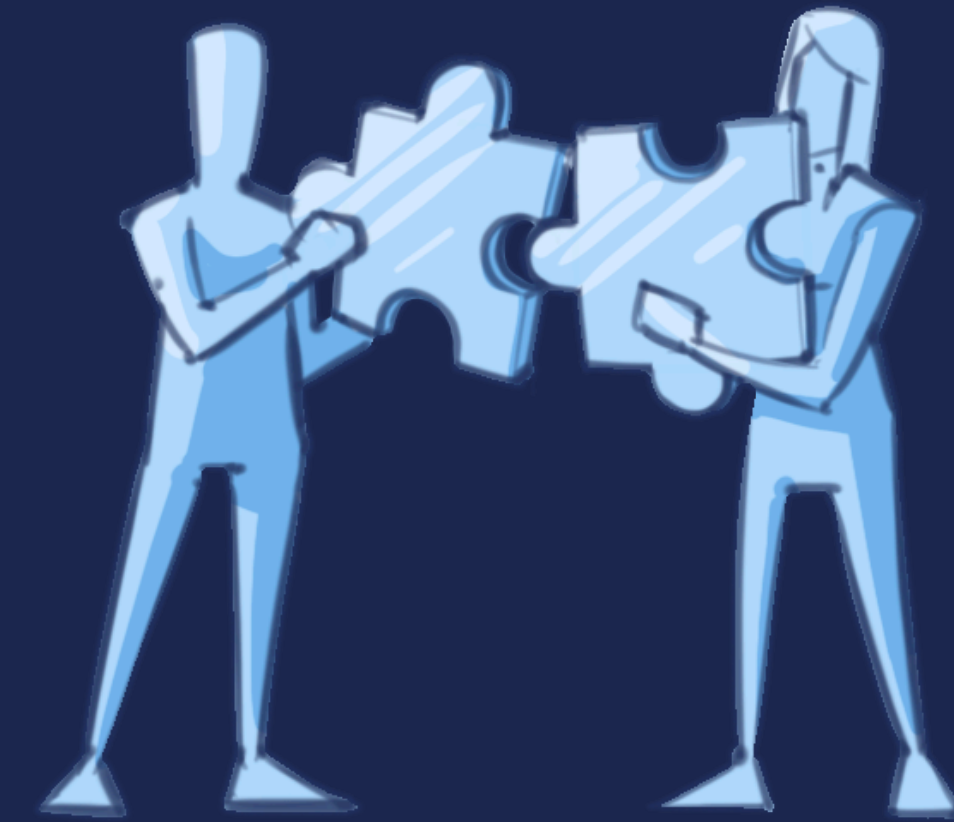

HOW TO GET STARTED 2/2

- ▶ Install **droid**
 - ▶ pip install **detect-droid**
 - ▶ <https://github.com/certeu/droid>
 - ▶ Make your own configuration file
 - ▶ <https://certeu.github.io/droid-docs>



ROADMAP

- ▶ Integrate more **platforms** (SIEM/EDR)
- ▶ Improve the **correlation** rules
- ▶ Finalise the **testing feature** using Atomic Red Team™
- ▶ Finalise the **documentation**





CONCLUSION

KEY TAKEAWAYS

- ▶ Using Sigma is essential for our work in a **multi-SIEM/EDR** environment
- ▶ We built a wrapper to ease our Sigma **implementation**
- ▶ Introducing **Detection-as-Code** to improve our service
 - ▶ ... quality assurance
 - ▶ ... content versioning
 - ▶ ... flexibility

REFERENCES

- ▶ Sigma

- ▶ <https://sigmahq.io/>
- ▶ <https://github.com/SigmaHQ/sigma>

- ▶ Detection-as-Code

- ▶ [Can We Have "Detection as Code"? - Anton Chuvakin](#)
- ▶ [From soup to nuts: Building a Detection-as-Code pipeline - David French](#)



Q&A



10 
years **CERT-EU**