Aurora

Your custom Sigma-based EDR Agent

What is Aurora?

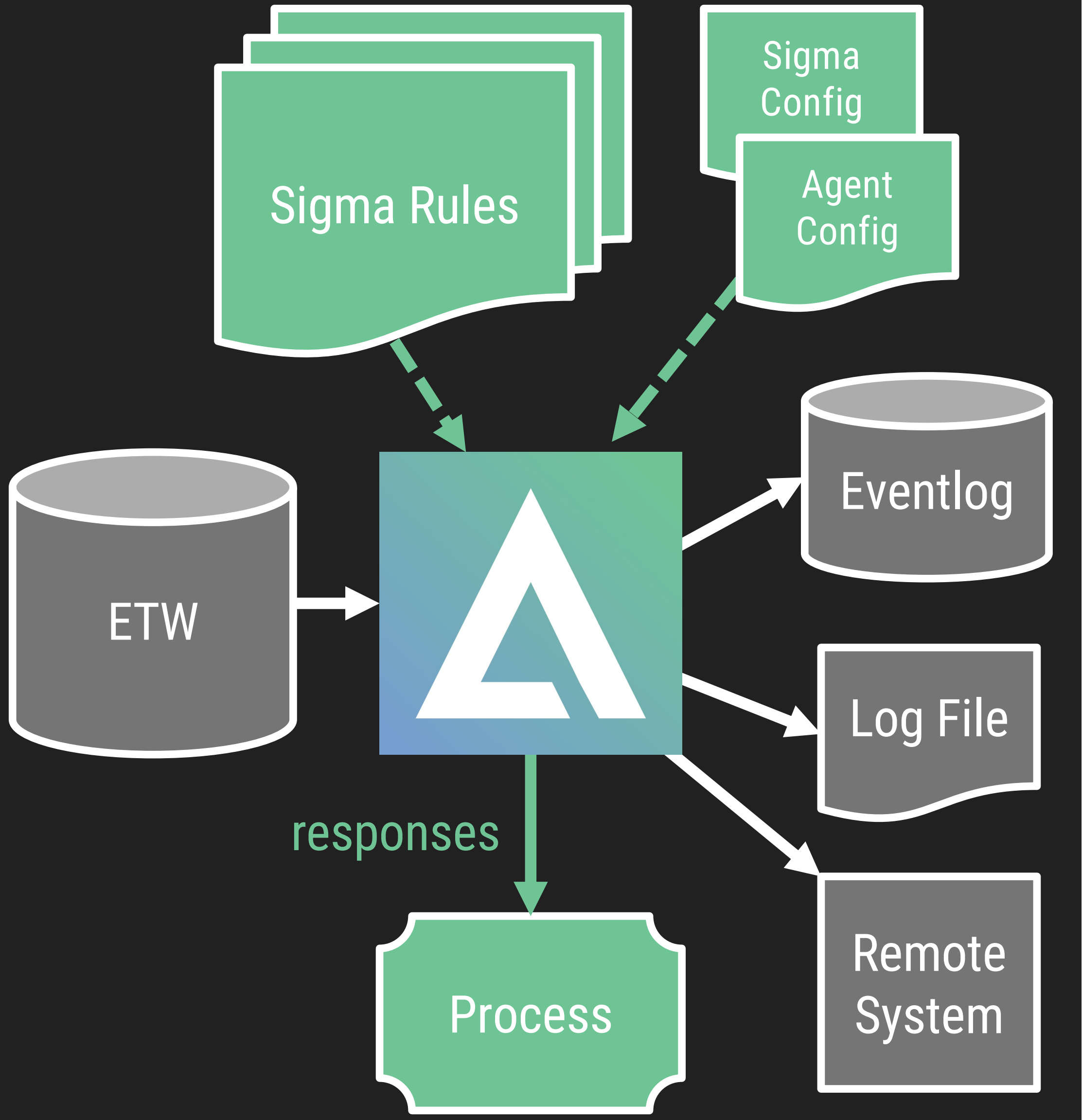
A lightweight agent
that applies Sigma rules
on endpoints

Aurora Agent – The Idea

- Lightweight agent that applies Sigma rules on log data in real-time on endpoints
- Uses ETW (Event Tracing for Windows)
- Managed locally via config files or via ASGARD Management Center
- Extends the Sigma standard with 'response' actions
  - Kill, KillParent, Suspend, Dump
  - Custom actions
- Supports the upcoming Sigma correlation rules
- Consider it your custom Sigma-based EDR
- Aurora Agent Lite
  - free, lacks comfort features and modules (e.g. Cobalt Strike beaconing detection)

Aurora Agent - Components

- **Agent Binary**
  the service binary that runs constantly and applies Sigma rules to monitored events
- **Sigma Rules**
  a directory with Sigma rules to apply
- **Sigma Config**
  a configuration file that includes mapping configuration for log sources and fields
- **Aurora Agent Config**
  a configuration file to set output options, log levels, configure rule sets etc.

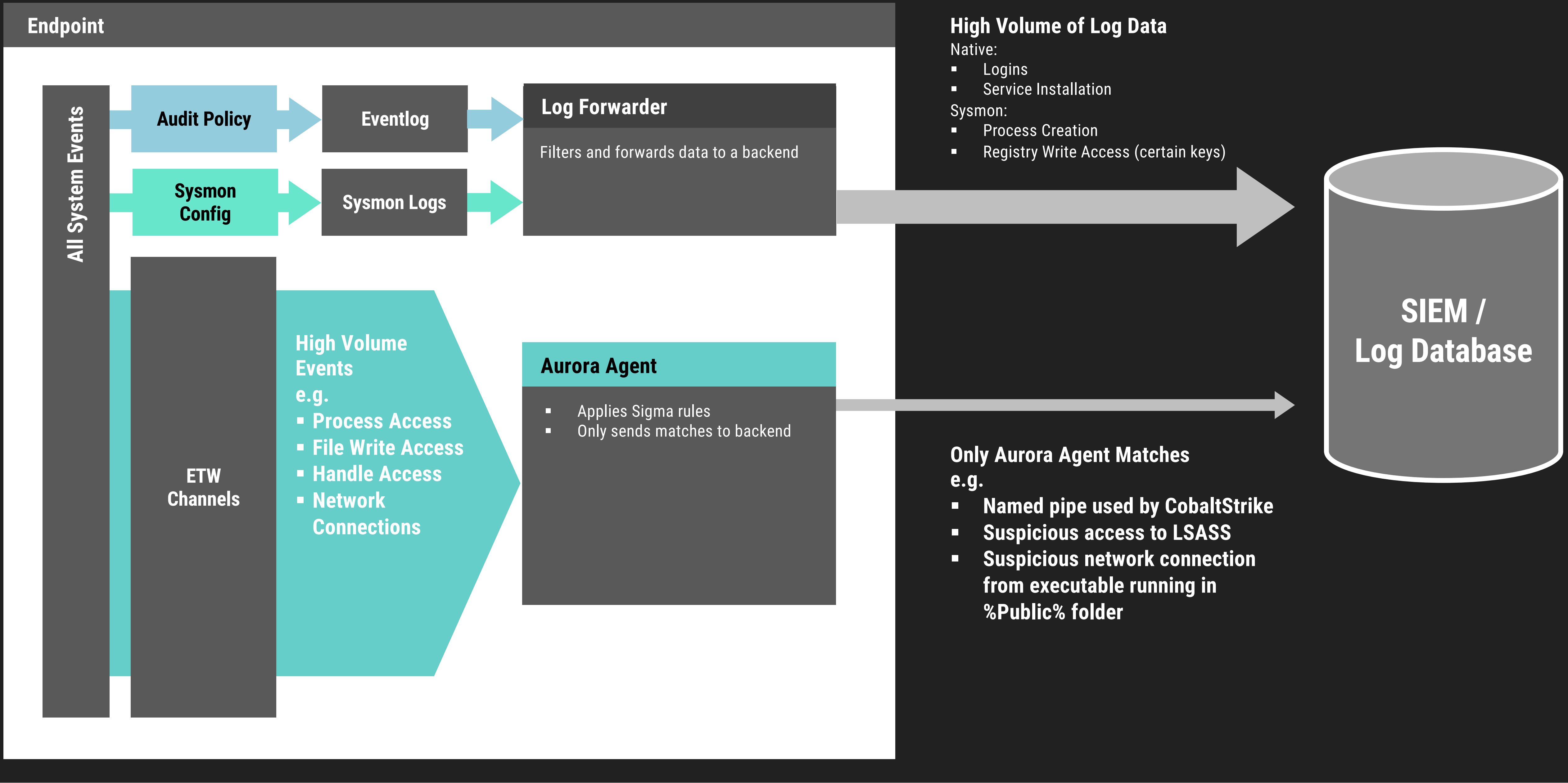


Advantages

Easily Customizable

- Sigma
  - open standard
  - many open source rules available (700+ for the Windows platform)
  - Our extension: response actions ⚡
- Add custom rules
  - from blog posts
  - write your own to detect or block custom threats (e.g. Ransomware containment)
  - from threat feeds (MISP, TI providers)

Reduced Log Volume



Endpoint

All System Events

Audit Policy

Eventlog

Sysmon Config

Sysmon Logs

Log Forwarder

Filters and forwards data to a backend

ETW Channels

High Volume Events e.g.

- Process Access
- File Write Access
- Handle Access
- Network Connections

Aurora Agent

- Applies Sigma rules
- Only sends matches to backend

High Volume of Log Data

Native:

- Logins
- Service Installation

Sysmon:

- Process Creation
- Registry Write Access (certain keys)

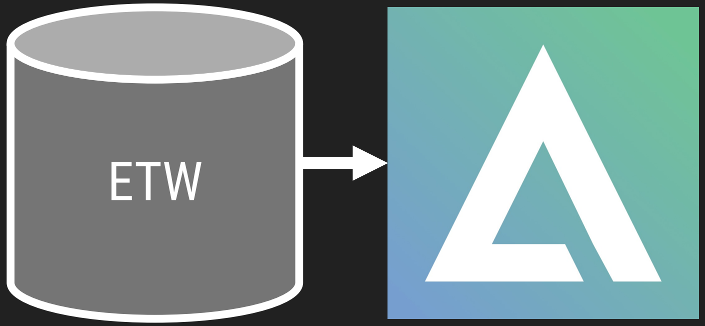
SIEM / Log Database

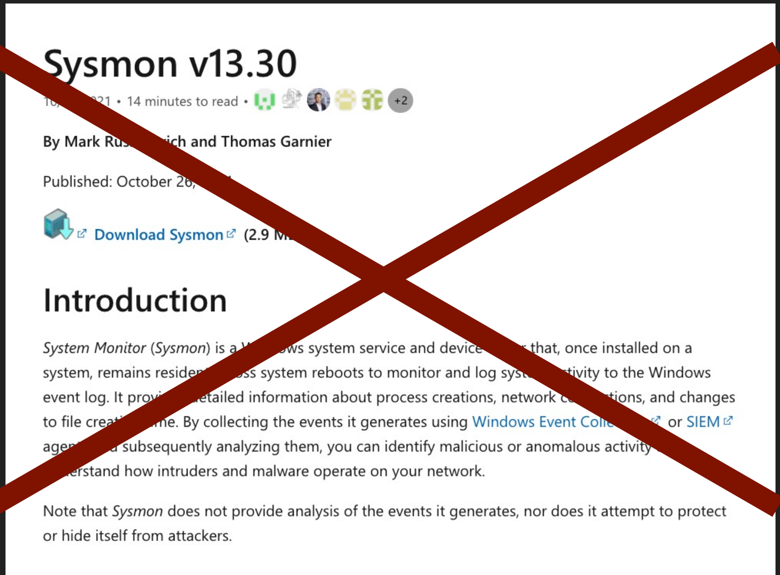
Only Aurora Agent Matches e.g.

- **Named pipe used by CobaltStrike**
- **Suspicious access to LSASS**
- **Suspicious network connection from executable running in %Public% folder**

Independence and Stability

- No specific Windows audit policy required
- No Sysmon required
- We tap into ETW, recreate 90%* of the events used in Sysmon and apply Sigma rules to them
- No Kernel Driver used (no blue screens)
  - Disadvantage: we miss some events (NamedPipe events, in some corner cases the CommandLine of a process)





Sysmon v13.30

10/[illegible]21 • 14 minutes to read •

By Mark Ru[illegible]ich and Thomas Garnier

Published: October 26, [illegible]

Download Sysmon (2.9 M[illegible]

Introduction

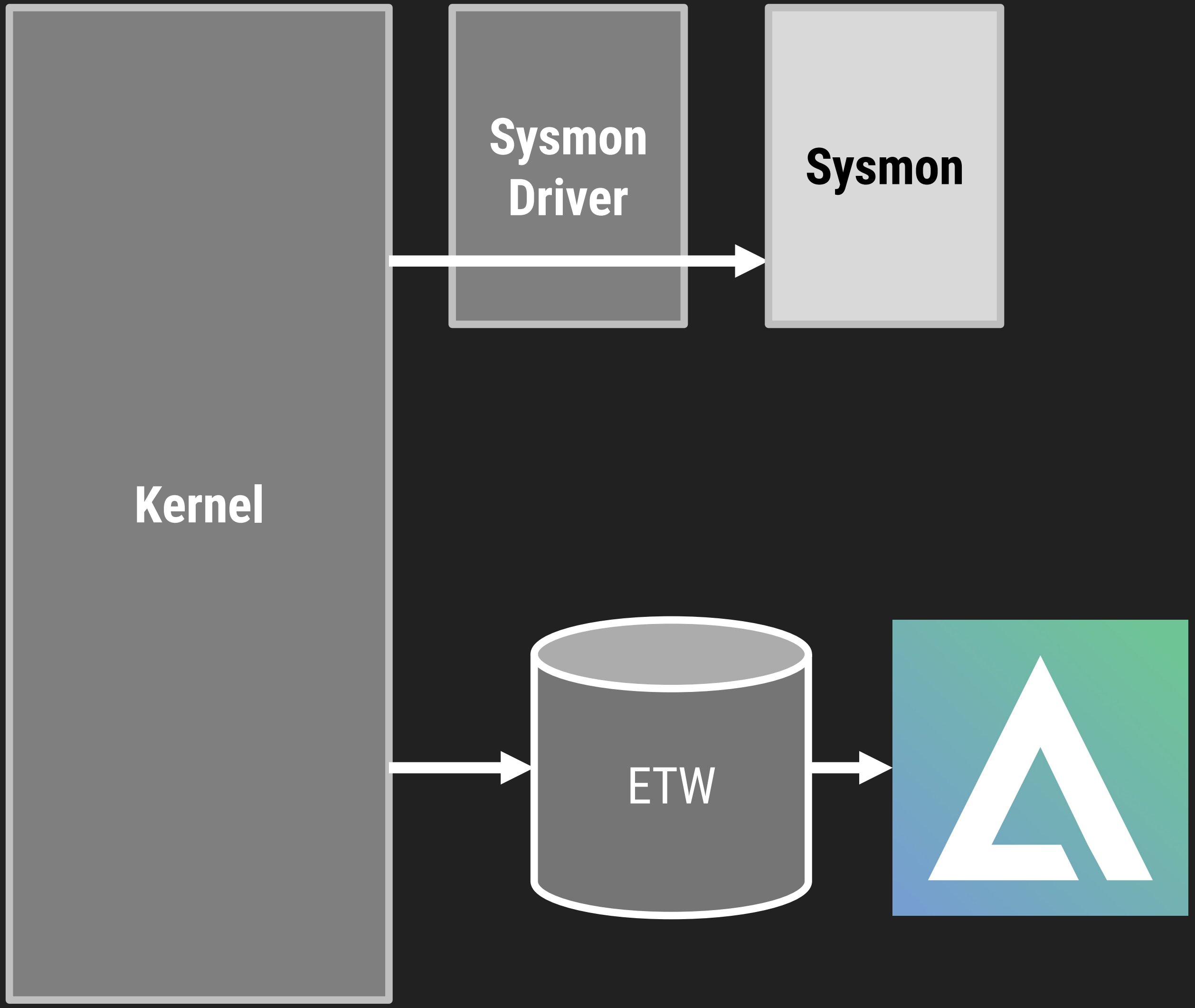
System Monitor (*Sysmon*) is a W[illegible]ows system service and device[illegible] that, once installed on a system, remains residen[illegible]ss system reboots to monitor and log sys[illegible]tivity to the Windows event log. It provi[illegible]etailed information about process creations, network c[illegible]tions, and changes to file crea[illegible]me. By collecting the events it generates using Windows Event Colle[illegible] or SIEM age[illegible] subsequently analyzing them, you can identify malicious or anomalous activity [illegible]erstand how intruders and malware operate on your network.

Note that *Sysmon* does not provide analysis of the events it generates, nor does it attempt to protect or hide itself from attackers.

*some event types & fields may not be available in the first release version, but the most important ones

Comparison to Sysmon

Recreation of Sysmon-like Events in Aurora



Event ID 1: Process Creation
- ProcessID
- Image
- ParentImage
- CommandLine
- Hash
- …

Event ID 2: A process changed a file creation time
Event ID 3: Network connection
Event ID 4: Sysmon service state change
Event ID 5: Process Terminated
Event ID 6: Driver loaded
- ImageLoaded
- Hashes
- Signature
- SignatureStatus

Event ID 1: Process Creation
- ProcessID
- Image
- ParentImage
- CommandLine
- Hash
- …

Event ID 2: A process changed a file creation time
Event ID 3: Network connection
Event ID 4: Sysmon service state change
Event ID 5: Process Terminated
Event ID 6: Driver loaded
- ImageLoaded
- Hashes
- Signature
- SignatureStatus

Percentage of Event / Fields	Percentage of Event / Fields used in Sigma Rules
~70%	~95%

Comparison Sysmon / Aurora

	Sysmon	Aurora
Event Source	Sysmon Kernel Driver	ETW (Event Tracing for Windows)
Sigma Rule Event Coverage	100%	95%
Relative Log Volume	High	Low
Sigma Matching	No	Yes
Response Actions	No	Yes
Resource Control (CPU Limiter)	No	Yes
Output: Eventlog	Yes	Yes
Output: File	No	Yes
Output: UDP target	No	Yes
Risk: Blue Screen	Yes	No
Risk: High System Load	Yes	No

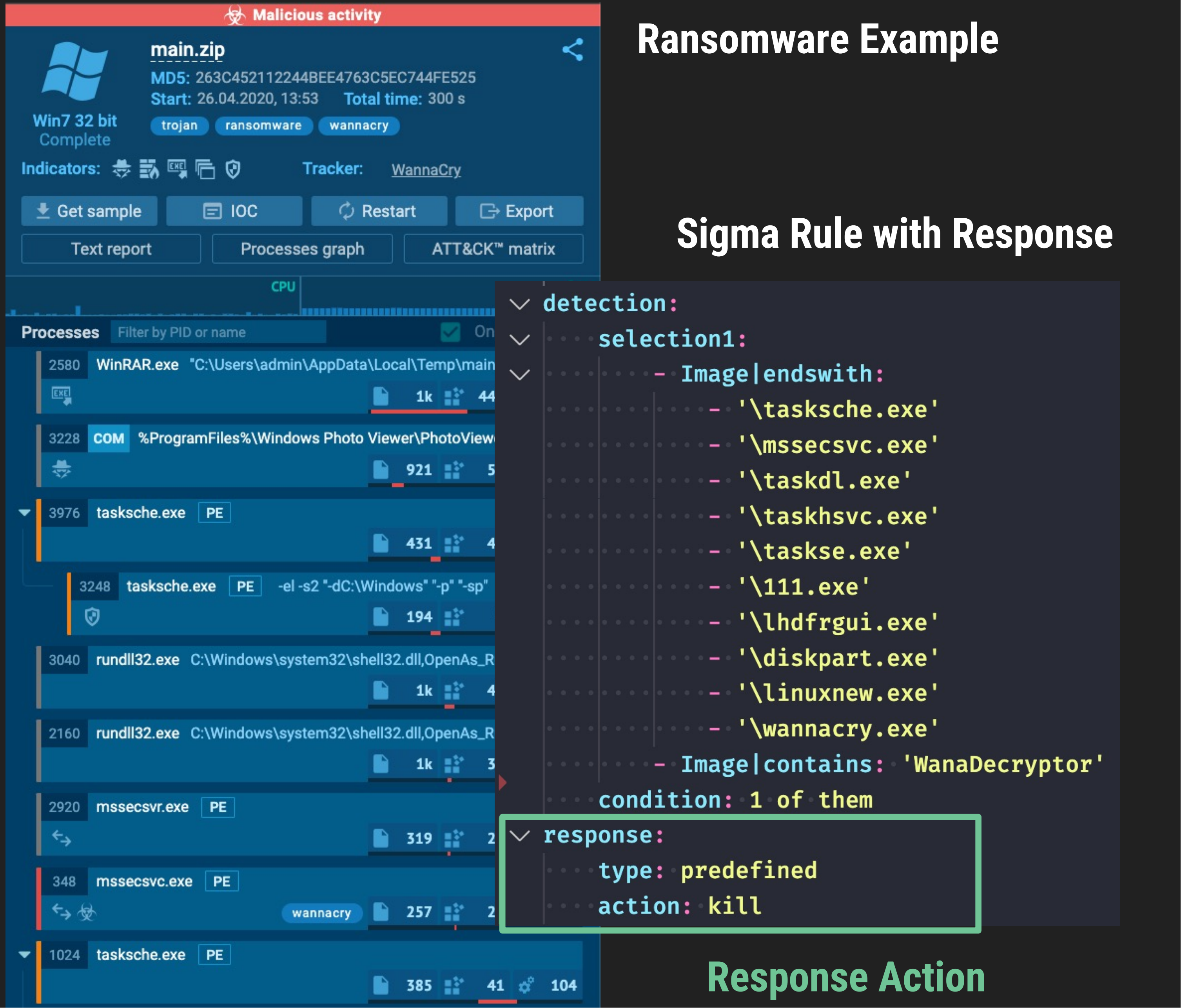
Response Actions

Response Actions

- Use Sigma to detect a threat
- Add a response action
  - Predefined
    - Kill a process or parent process
    - Suspend a process
    - Dump process memory
  - Custom
    - A custom command line that can make use of environment variables and the event's values
      e.g. copy %Image% %%ProgramData%%\%ProcessId%.bin
- Contain threats in milliseconds

Ransomware Example

Sigma Rule with Response



Malicious activity

main.zip
MD5: 263C452112244BEE4763C5EC744FE525
Start: 26.04.2020, 13:53 Total time: 300 s
trojan ransomware wannacry
Win7 32 bit
Complete
Indicators:
Tracker: WannaCry

```
2580  WinRAR.exe  "C:\Users\admin\AppData\Local\Temp\main
                                    1k   44
3228  COM  %ProgramFiles%\Windows Photo Viewer\PhotoView
                                    921  5
3976  taskshe.exe  PE
                                    431  4
   3248  tasksche.exe  PE  -el -s2 "-dC:\Windows" "-p" "-sp"
                                    194
3040  rundll32.exe  C:\Windows\system32\shell32.dll,OpenAs_R
                                    1k   4
2160  rundll32.exe  C:\Windows\system32\shell32.dll,OpenAs_R
                                    1k   3
2920  mssecsvr.exe  PE
                                    319  2
348   mssecsvc.exe  PE
                         wannacry   257  2
1024  tasksche.exe  PE
                                    385  41  104
```

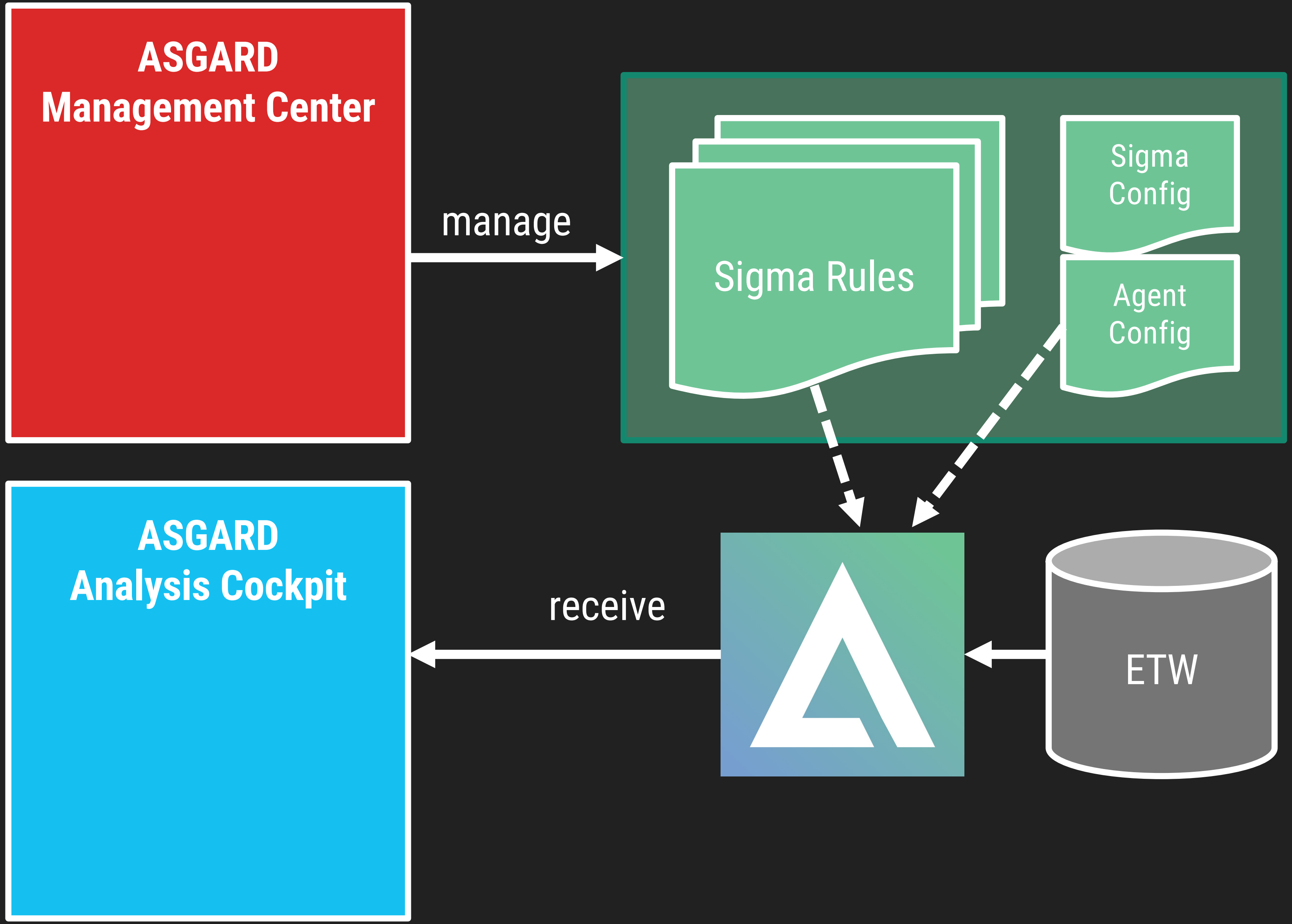
```
detection:
    selection1:
        - Image|endswith:
            - '\tasksche.exe'
            - '\mssecsvc.exe'
            - '\taskdl.exe'
            - '\taskhsvc.exe'
            - '\taskse.exe'
            - '\111.exe'
            - '\lhdfrgui.exe'
            - '\diskpart.exe'
            - '\linuxnew.exe'
            - '\wannacry.exe'
        - Image|contains: 'WanaDecryptor'
    condition: 1 of them
response:
    type: predefined
    action: kill
```

Response Action

ASGARD Integration

ASGARD and Aurora Agent

- Comfortable Sigma rule management
  - Enable / disable rules
  - Create rule sets for different asset groups
  - Get rule updates from public sources
  - Identify changes in updated rules and decide to deploy them
- Use Nextron's private Sigma rules
- Analyse and base-line Sigma rule matches



Aurora Lite and Aurora Differences

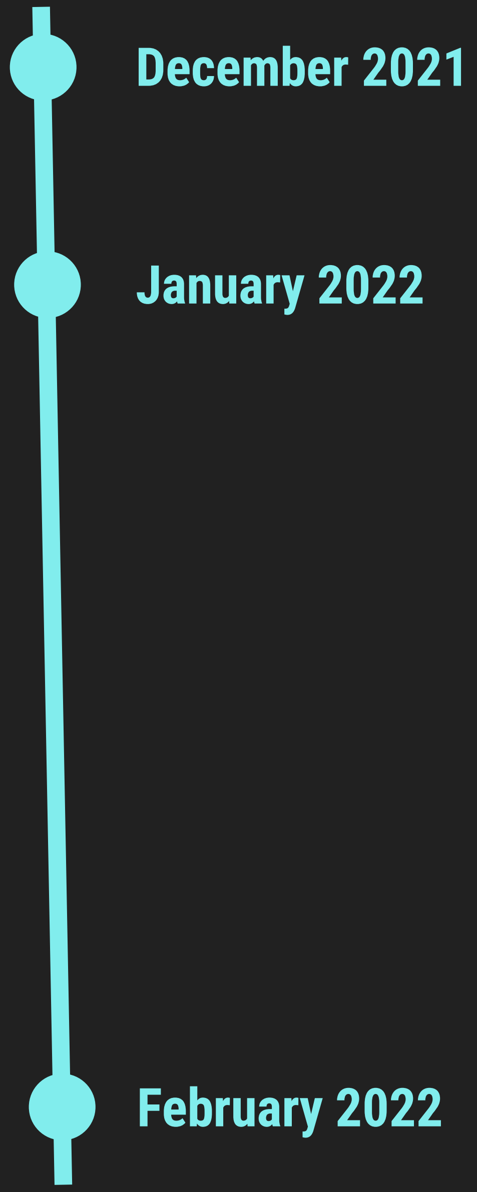
Aurora Features

Not included / possible in the light version

1. ASGARD Integration
   - Simple agent deployment
   - Comfortable rule management and deployment
   - Managed rule and agent updates
2. Non-Sigma Based Threat Detection Modules
   - Cobalt Strike beaconing detection module
   - LSASS process dump detection module
   - there are more to come
3. Nextron’s private Sigma rule feed
4. Encrypted Sigma rules

Aurora Roadmap

- Release Candidate 1 (RC1)
- Github repository for Sigma rules meant to be used in Aurora (with response actions)
- Self-protection measures
- More non-Sigma-based modules
- More predefined response actions
  - “EMP”: kill all processes in a certain user context that have been created X seconds before an event
  - ”isolate”: modify routing table / firewall of system so that it can only communicate with a certain host
  - “yara”: scan a process memory / image with YARA and rules in folder X
- Linux version (eBPF)



December 2021

January 2022

February 2022

Get Started

Visit the contact form an mention “Aurora Agent”
https://www.nextron-systems.com/get-started/