

Performing enterprise-wide DFIR and Threat Hunting with Yamato Security OSS Tools

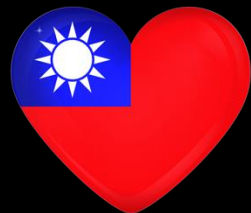
Zach Mathis, Akira Nishikawa, Fukusuke Takahashi

2024/08/23



MIND MELO HACKER SPIRIT FROM HUMAN TO AI

Congratulations on 20 years!!!
20周年、おめでとうございます！



Zach Mathis

- Yamato Security founder and project leader
- 2006~ Kobe Digital Labo (KDL)
- 2016~ SANS 504 Instructor

Akira Nishikawa

- First core developer
- Kaminashi
- 2007~ Freelance engineer
- 2021~ SaaS product security
- AWS Community Builder

Fukusuke Takahashi

- Latest core developer
- NTTDATA-CERT
- DFIR, OSINT, SOAR
- Fixes OSS tool bugs and does bug hunting in free time

Agenda

- About Yamato Security and tools/resources
- Scalable DFIR built with Velociraptor & Hayabusa
- Easy analysis with Takajo
- Open-source tool management
- Future plans

About Yamato Security tools and resources

About Yamato Security



- A Japanese security group that provides free/cheap training since 2012.
- Develops various open-source security tools and resources since 2021:



Project Leader

Zach Mathis (@yamatosecurity)

Developers

Akira Nishikawa (@nishikawaakira)

DustInDark / hitenoku

James Takai / hachiyone(@hach1yon)

ItiB (@itiB_S144)

Kazuminn (@k47_um1n)

Garigariganzy (@garigariganzy31)

Fukusuke Takahashi / fukuseket

Yusuke Matsui (@apt773)(AD Hacking Group Leader)



Project Leader

Zach Mathis (@yamatosecurity)

Developers

Akira Nishikawa (@nishikawaakira)

DustInDark / Hitenoku

Fukusuke Takahashi / fukusuket



Follow:

@SecurityYamato

**Who has used Yamato
Security tools (Hayabusa,
Takajo, RustyBlue, WELA,
etc...) before?
(Please raise your hand)**

Yamato Security tools and resources

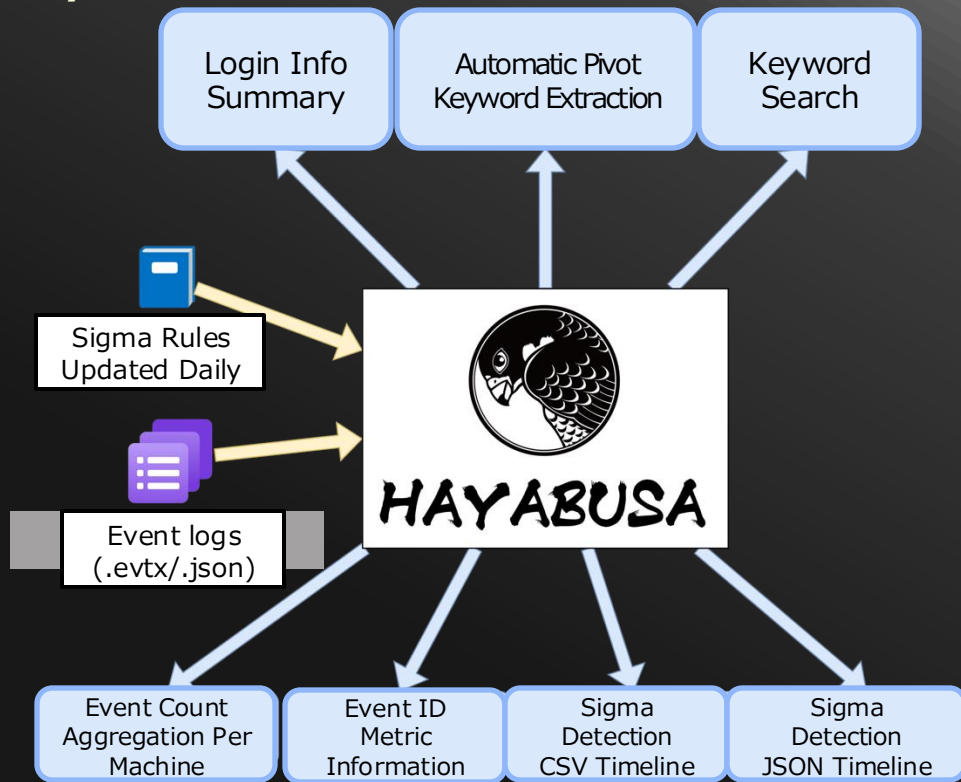
- **Hayabusa**: DFIR timeline generator using native Sigma rules for Windows event logs
- **Takajo**: Hayabusa results analyzer
- Yamato Security's Windows Event Log Configuration Guide For DFIR And Threat Hunting
- Curation of Sigma Rules for Windows Event Logs
- **Deprecated**: WELA

Hayabusa



<https://github.com/Yamato-Security/hayabusa>

- Fast forensics and Threat Hunting tool
- Quickly analyzes large amounts of Windows event logs.
- Written in Rust so it is very fast and cross-platform and safe from anti-forensics memory corruption exploits
- Has various commands specialized for Windows event log analysis
- Synchronize with Sigma repository with one command. Over 4000 Sigma and built-in detection rules to output a CSV or JSON timeline!



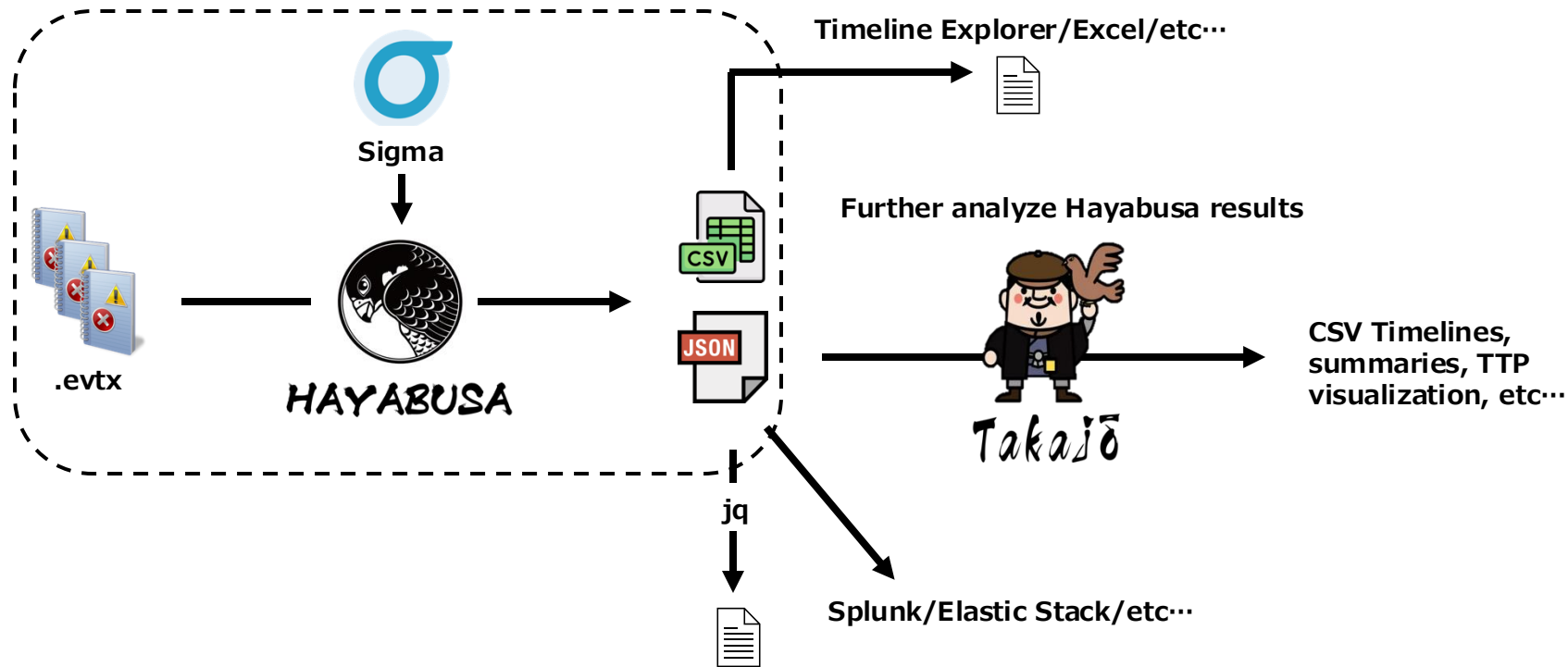
Sigma



<https://github.com/SigmaHQ/sigma>

- Free open source detection rules for various platforms (Windows event logs, cloud, linux, etc···)
- 4000+ rules for Windows event logs
- Advantages:
 - Easy to write/read YAML files
 - Many free, high-quality and up-to-date rules
 - Can convert to any backend out there (Splunk, Elastic Stack, Qradar, KQL, etc···)

Hayabusa overview



Hayabusa



<https://github.com/Yamato-Security/hayabusa>

- Best native support for Sigma rules! Hayabusa supports all major field modifiers including the Sigma correlation rules that are still in development.
- Contributors: 14
- Releases / Major updates: 45!
- Major updates almost monthly for almost 3 years!
- Commits: 4200+!
- Issues closed: 500+
- Pull Requests Merged: ~800
- Downloads: 113,000+!
- Hayabusa detection rules: ~200
- Sigma detection rules: 4000+
- Talks: Black Hat, CODE BLUE, SANS DFIR Summit, Bsides Tokyo, Hack Fes, FIRSTCON, etc...
- Used world-wide by many CERTs and DFIR specialists!

Hayabusa 2.17.0 Release!



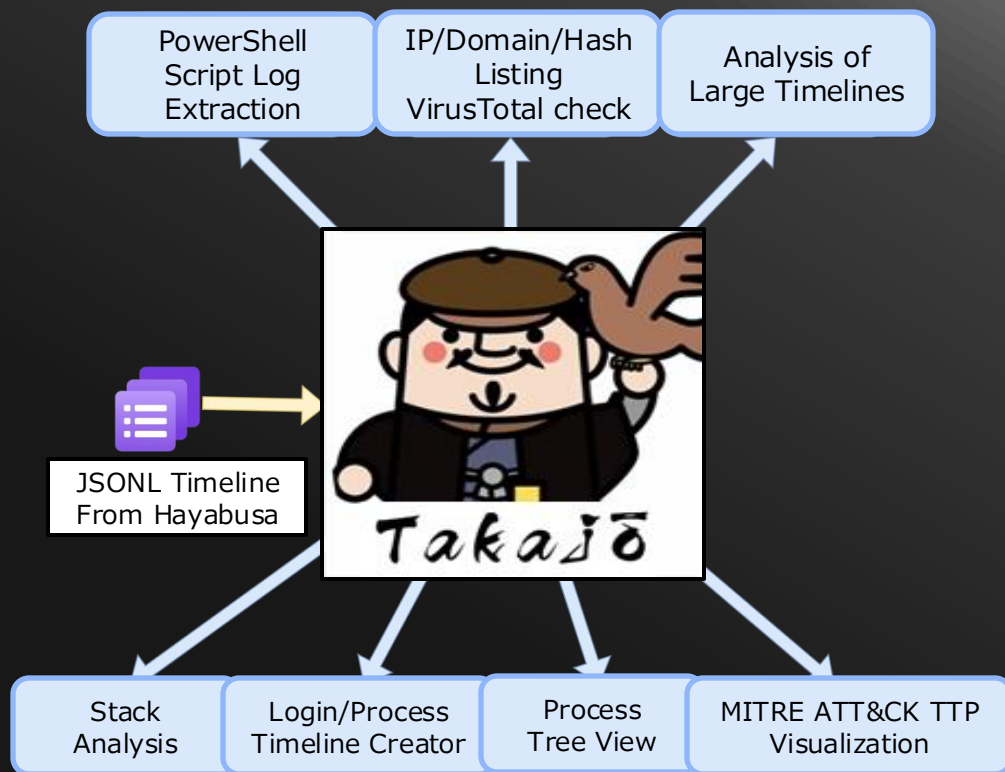
<https://github.com/Yamato-Security/hayabusa>

- We are releasing the latest version 2.17.0 “HITCON Community Release” today!
- Low memory mode is enabled by default so you can scan 100GB+ of event logs with less than 4GB of memory!
- Better support for Sigma correlation rules
- Aggregation rules show more information like Event ID and Channel
- The “windash” modifier has been updated to detect special dash characters used to bypass signatures
- Various bug fixes
- Download here: <https://github.com/Yamato-Security/hayabusa/releases>
- Contributors: Fukusuke Takahashi and DustInDark (hitenkoku)

Takajo

<https://github.com/Yamato-Security/takajo>

- Written in Nim so is easy to program as python but is as fast as C !
- Main developers: DustInDark, Zach Mathis, Fukusuke Takahashi
- New version 2.6.0 "HITCON Community Release" today!
- New "html-report" web report command thanks to Akira Nishikawa!
- We will explain more in detail later in this presentation



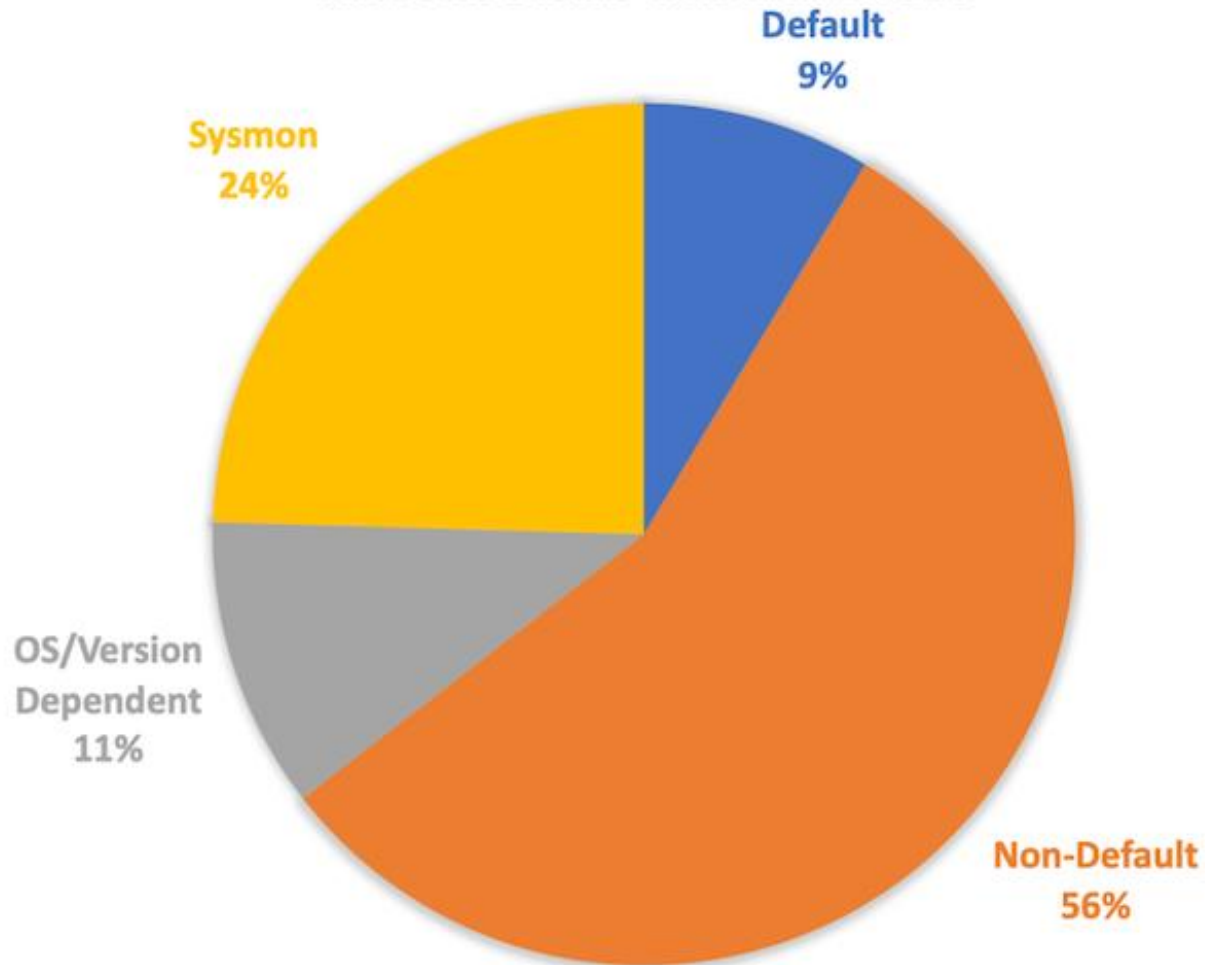
Yamato Security's Windows Event Log Configuration Guide For DFIR And Threat Hunting



[**https://github.com/Yamato-Security/EnableWindowsLogSettings**](https://github.com/Yamato-Security/EnableWindowsLogSettings)

- Yet another guide for properly configuring Windows event log audit settings
- Very practical because it is based on real world Sigma detection rules
- Tells you what attacks you are able to detect if you enable certain settings
- Visualization of the most important Windows audit settings
- Includes a batch script to automatically configure your Windows audit settings correctly!
- Documentation: Zach Mathis and Fukusuke Takahashi

WINDOWS EVENTS WITH SIGMA RULES



Sigma Log Source ▾	Channel and EID ▾	Default Settings ▾	Rules ▾	Percent ▾
process_creation	Microsoft-Windows-Sysmon/Operational 1 or Security 4688	non-default	804	49.36%
security	Security	partial	139	8.53%
ps_script	Microsoft-Windows-PowerShell/Operational 4104	partial	125	7.67%
registry_set	Microsoft-Windows-Sysmon/Operational 13	sysmon	109	6.69%
file_event	Microsoft-Windows-Sysmon/Operational 11	sysmon	96	5.89%
system	System	default	50	3.07%
image_load	Microsoft-Windows-Sysmon/Operational 7	sysmon	39	2.39%
registry_event	Microsoft-Windows-Sysmon/Operational 12/13/14	sysmon	37	2.27%
ps_module	Microsoft-Windows-PowerShell/Operational 4103	non-default	30	1.84%
network_connection	Microsoft-Windows-Sysmon/Operational 3	sysmon	29	1.78%
process_access	Microsoft-Windows-Sysmon/Operational 10	sysmon	25	1.53%
pipe_created	Microsoft-Windows-Sysmon/Operational 17/18	sysmon	14	0.86%
application	Application	default	13	0.80%
dns_query	Microsoft-Windows-Sysmon/Operational 22	sysmon	12	0.74%
ps_classic_start	Windows PowerShell 400	default	10	0.61%
create_remote_thread	Microsoft-Windows-Sysmon/Operational 8	sysmon	10	0.61%

Curation of Sigma Rules for Windows Event Logs and Hayabusa Rules Repo



<https://github.com/Yamato-Security/sigma-to-hayabusa-converter>

<https://github.com/Yamato-Security/hayabusa-rules>

- Documentation, research, conversion design: Zach Mathis
- Initial Python implementation: James Takai and Itib
- Research, current implementation: Fukusuke Takahashi
- Sigma rule "logsource" field is de-abstracted to concrete field names in new rules so it is easier to understand the rule and we can now support many built-in event logs in case Sysmon cannot be installed:
- Process Creation (Sysmon 1 => Security 4688)
- Registry Events (Sysmon 12, 13, 14) => Security 4657)
- Network Events (Sysmon 3 => Security 5156)
- New rules are curated and saved to the hayabusa-rules repository

Original Sigma Rule

```
logsource:  
  category: process_creation  
  product: windows  
detection:  
  selection:  
    - Image|endswith: '.exe'  
  condition: selection
```



Sysmon Rule

```
logsource:  
  category: process_creation  
  product: windows  
detection:  
  process_creation:  
    Channel: Microsoft-Windows-Sysmon/Operational  
    EventID: 1  
  selection:  
    - Image|endswith: '.exe'  
  condition: process_creation and selection
```

Built-In Event Rule

```
logsource:  
  category: process_creation  
  product: windows  
detection:  
  process_creation:  
    Channel: Security  
    EventID: 4688  
  selection:  
    - NewProcessName|endswith: '.exe'  
  condition: process_creation and selection
```

Sysmon 1	Sysmon 1 Example	Security 4688	Security 4688 Example					
User	DOMAIN\User	SubjectUserName	User					
		SubjectDomainName	DOMAIN					
LogonId	0x1864E	SubjectLogonId	0x1864e					
ProcessId	2468	NewProcessId	0x9a4					
Image	C:\Windows\System32\PING.EXE	NewProcessName	C:\Windows\System32\PING.EXE					
ParentProcessId	7772	ProcessId	0x1e5c					
CommandLine	C:\WINDOWS\system32\PING.EXE 8.8.8.8	CommandLine	"C:\WINDOWS\system32\PING.EXE" 8.8.8.8					
ParentImage	C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe	ParentProcessName	C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe					
IntegrityLevel	High	MandatoryLabel	S-1-16-12288					
		TargetUserSid	S-1-0-0					
		TargetUserName	-					
		TargetDomainName	-					
		TargetLogonId	0x0					
		SubjectUserSid	S-1-5-21-2977773840-2930198165-1551093962-1000					
		TokenElevationType	%1937					
RuleName		<div>Author: Zach Mathis (@yamatosecurity)</div> <div><table><tr><th>Legend</th></tr><tr><td>Exists in both events</td></tr><tr><td>Needs field name conversion</td></tr><tr><td>Needs field name and value conversion</td></tr><tr><td>Only exists in one event</td></tr></table></div>		Legend	Exists in both events	Needs field name conversion	Needs field name and value conversion	Only exists in one event
Legend								
Exists in both events								
Needs field name conversion								
Needs field name and value conversion								
Only exists in one event								
UtcTime	2019-06-15 07:13:42.278							
ProcessGuid	{365ABB72-9AA6-5D04-0000-00109C850F00}							
FileVersion	10.0.19041.1 (WinBuild.160101.0800)							
Description	TCP/IP Ping Command							
Product	Microsoft® Windows® Operating System							
Company	Microsoft Corporation							
OriginalFileName	ping.exe							
CurrentDirectory	C:\tools\Sysmon-15\							
LogonGuid	{365ABB72-98E4-5D04-0000-0020A4350100}							
TerminalSessionId	1							
Hashes	SHA1=D4F0397F83083E...							
ParentProcessGuid	{365ABB72-9972-5D04-0000-0010F0490C00}							
ParentCommandLine	C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe							
ParentUser	DOMAIN\User							

Sysmon 3	Sysmon 3 Example	Security 5156	Security 5156 Example
ProcessId	3080	ProcessID	3080
Image	C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe	Application	\device\harddiskvolume4\windows\system32\windowspowershell\v1.0\powershell.exe
Protocol	tcp	Protocol	6
Initiated	true	Direction	%%14593
SourceIp	10.0.0.4	SourceAddress	10.0.0.4
SourcePort	49775	SourcePort	49775
DestinationIp	93.184.215.14	DestAddress	93.184.215.14
DestinationPort	443	DestPort	443
RuleName	-	<div>Author: Fukusuke Takahashi (@fukusuket)</div> <div> <div>Legend</div> <div>Exists in both events</div> <div>Needs field name conversion</div> <div>Needs field name and value conversion</div> <div>Only exists in one event</div> </div>	
UtcTime	45501.34839		
ProcessGuid	{09e2f3ec-ff78-66a5-d700-000000000500}		
DestinationHostname	-		
DestinationPortName	https		
SourceIsIpv6	false		
SourceHostname	samurai.g514.mx.internal.cloudapp.net		
SourcePortName	-		
DestinationIsIpv6	false		
User	samurai\samurai		
		FilterRTID	68840
		LayerName	%%14611
		LayerRTID	48
		RemoteUserID	S-1-0-0
		RemoteMachineID	S-1-0-0

Curation of Sigma Rules for Windows Event Logs and Hayabusa Rules Repo



<https://github.com/Yamato-Security/sigma-to-hayabusa-converter>

<https://github.com/Yamato-Security/hayabusa-rules>

- Benefits of de-abstracting “logsource” field:
 - Can detect attacks in built-in Windows events
 - Easier to read and understand the rule
 - Easier for native Sigma-parsers to use
- For these reasons, the curated rules in the hayabusa-rules repository are used for Hayabusa and Velociraptor.
- Please read the (long) documentation for all of the details!

Scalable DFIR built with Velociraptor & Hayabusa

Environments where EDR and SIEM operations are insufficient



Environments where EDR and SIEM operations are insufficient



**Insufficient
log collection**

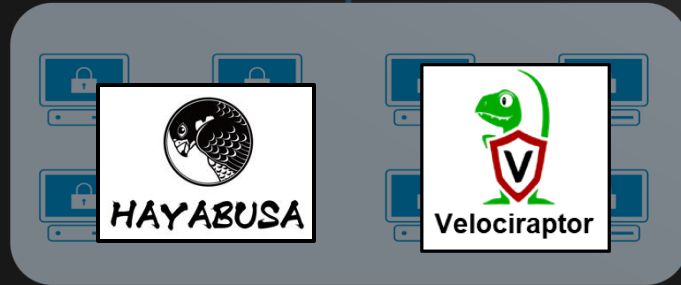
**Missing agent
settings**

**Rules not
detecting
attacks**

**Ignored
alerts**



Scalable DFIR with Velociraptor, Hayabusa and Takajo!



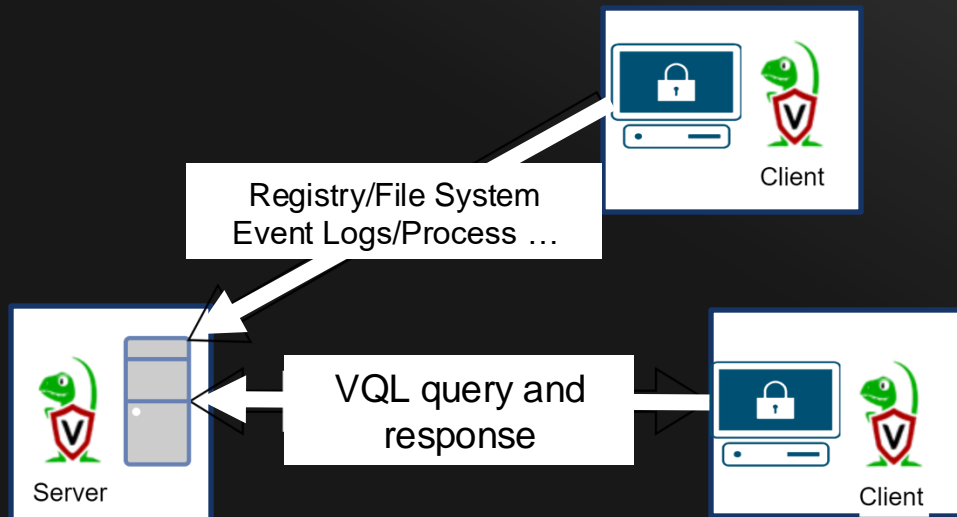
- Go back in time and recreate a SIEM!
- Fast forensics on many machines!

About Velociraptor



<https://docs.velociraptor.app>

- An OSS EDR-like DFIR tool developed by Mike Cohen and Rapid 7.
- Very good at collecting and analyzing forensic artifacts.
- Each artifact is gathered by executing “VQL” queries from server to client.
- Leverage knowledge by importing queries (VQL) shared by the community.



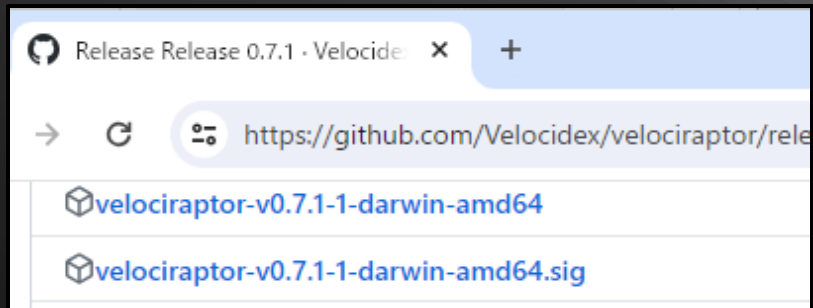
Velociraptor environment setup



Set up in 3 easy steps:

1. Server module startup
 - velociraptor gui
2. Create MSI package for distribution
 - Run `Server.Utils.CreateMSI` artifact
3. Distribute to client terminal
 - Distribute the MSI package created in <Step 2> using AD Group Policy, Intune, EDR, etc.

... And the setup is now complete!



Velociraptor Server Web GUI

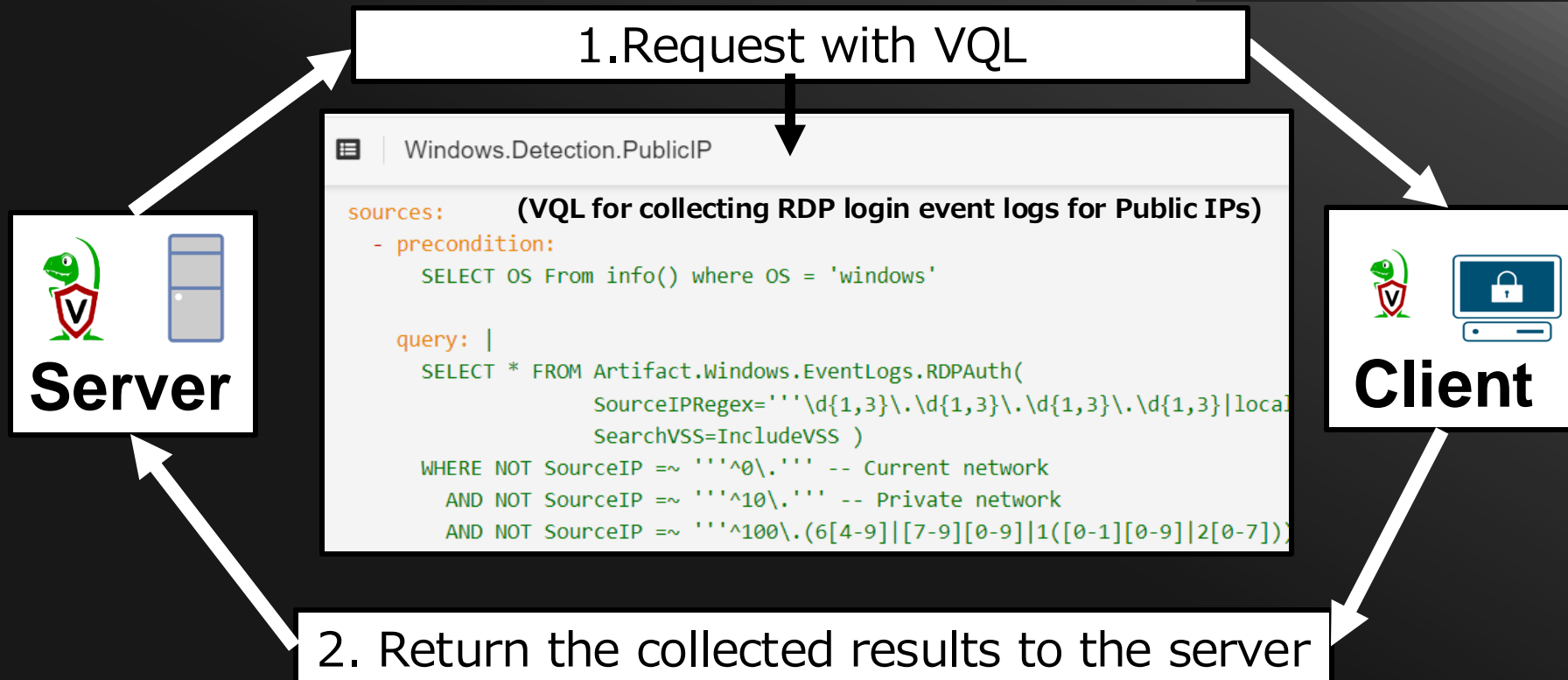


The screenshot displays the Velociraptor Server Web GUI interface. At the top, there is a search bar labeled "Search clients" with a magnifying glass icon and a dropdown arrow. Below the search bar, there are three icons: a home icon, a refresh icon, and a delete icon. A green status bar indicates "Query: all" and "Total Matching Clients 1".

<input type="checkbox"/>	<input type="radio"/>	Client ID	Hostname	FQDN	OS Version
<input type="checkbox"/>	<input checked="" type="radio"/>	C.632a4ab99b91b03b	mouse	mouse	Microsoft Windows 11 Home10.0.2263

At the bottom, there are pagination controls showing "10", "25", "30", and "50" items per page. On the right, there are navigation buttons: "«", "0", "»", and a "Goto Page" input field.

Velociraptor Basics

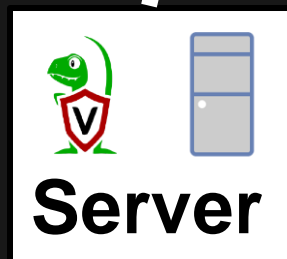


Distribution, execution, and collection of Hayabusa using VQL



1. Request Hayabusa distribution + execution with VQL

<https://docs.velociraptor.app/exchange/artifacts/pages/windows.eventlogs.hayabusa/>



```
Windows.EventLogs.Hayabusa

name: Windows.EventLogs.Hayabusa
description: |
  [Hayabusa](https://github.com/Yamato-Security/hayabusa) is a
  Windows event log fast forensics timeline generator and threat
  intelligence tool.

tools:
- name: Hayabusa-2.13.0
  url: https://github.com/Yamato-Security/hayabusa/releases/download/v2.13.0/hayabusa-2.13.0-win-x64.exe
  expected_hash: c350ba83ffb02391115d2d5e1236a6a1cd79e9c49be8296f485819e7b20be8fa
  version: 2.13.0

precondition: SELECT OS From info() where OS = 'windows'
```



2. Return the collected results to the server

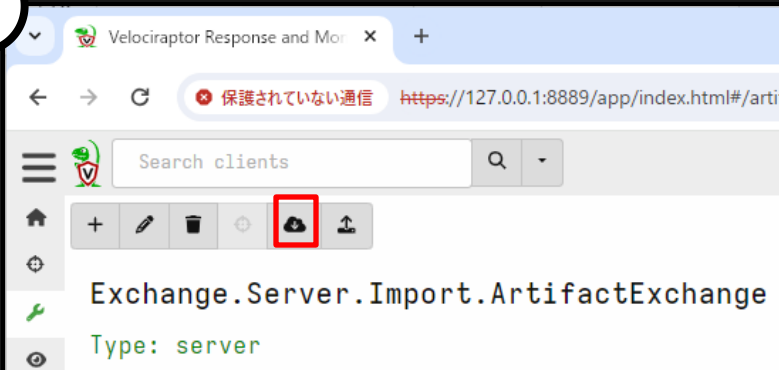
Distribution, execution, and collection of Hayabusa using VQL



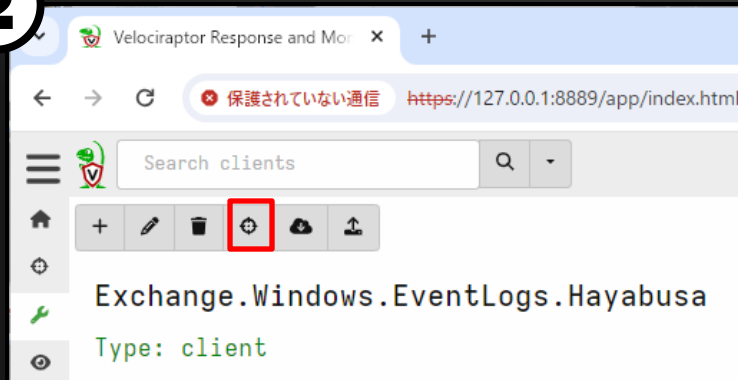
Can be started in 2 easy steps!

1. Run `Server.Import.ArtifactExchange` to import the community VQL
2. Run `Exchange.Windows.EventLogs.Hayabusa` !

1



2



Collecting Windows event log timeline with Hayabusa



Artifact Collection | Uploaded Files | Requests | **Results** | Log | Notebook

Exchange.Windows.EventLogs.Hayabusa/Results

Hayabusa's results
(Windows event logs scanned with Sigma rules)

Timestamp	RuleTitle	Level	Computer	Channel	EventID	RecordID	Details
2023-10-13T04:06:52Z	Important Log File Cleared	high	DESKTOP-CNG7416	Sys	104	2645	Log: System User: defaultuser0
2023-10-15T03:56:01Z	Windows Defender Real-time Protection Disabled	high	mouse	Defender	5001	99	Product Name: Microsoft Defender ウィルス対策 Product Version: 4.18.2201.11

The next step...

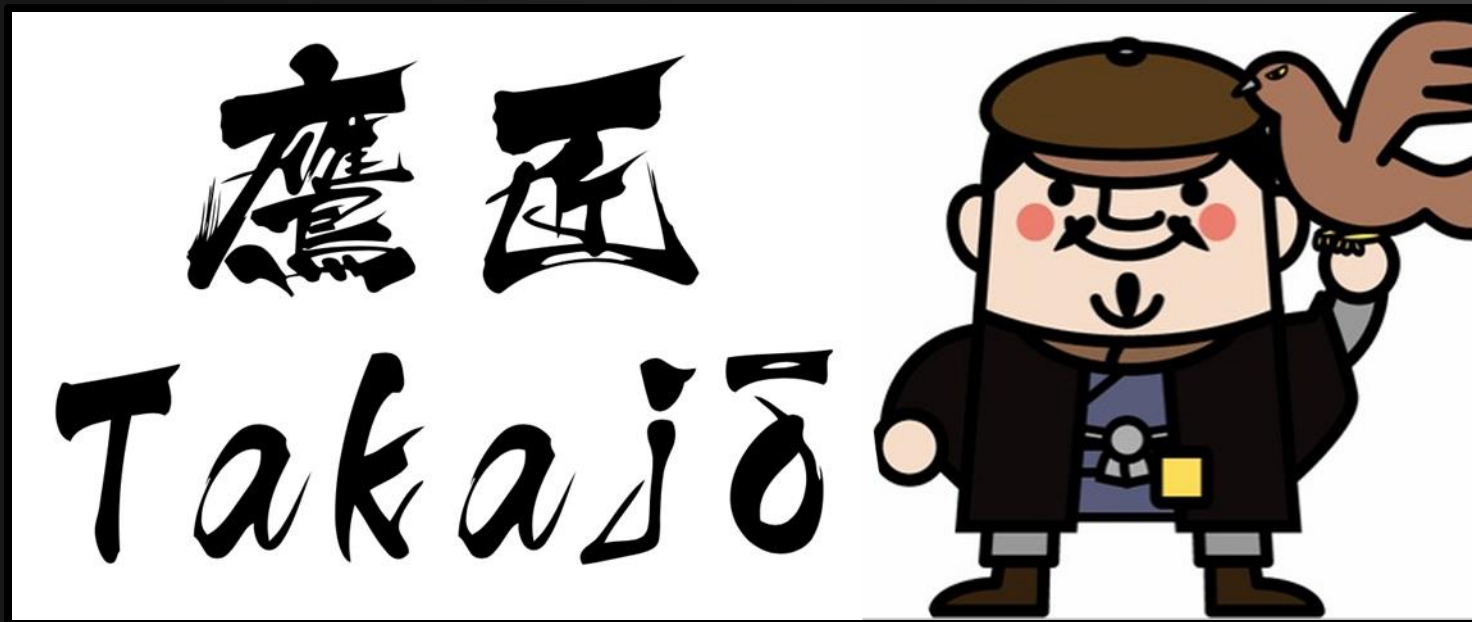


We collected event log analysis results
from multiple clients at once using
Velociraptor and Hayabusa!

The next step is to analyze this all at once 🤔

Easy analysis with Takajo

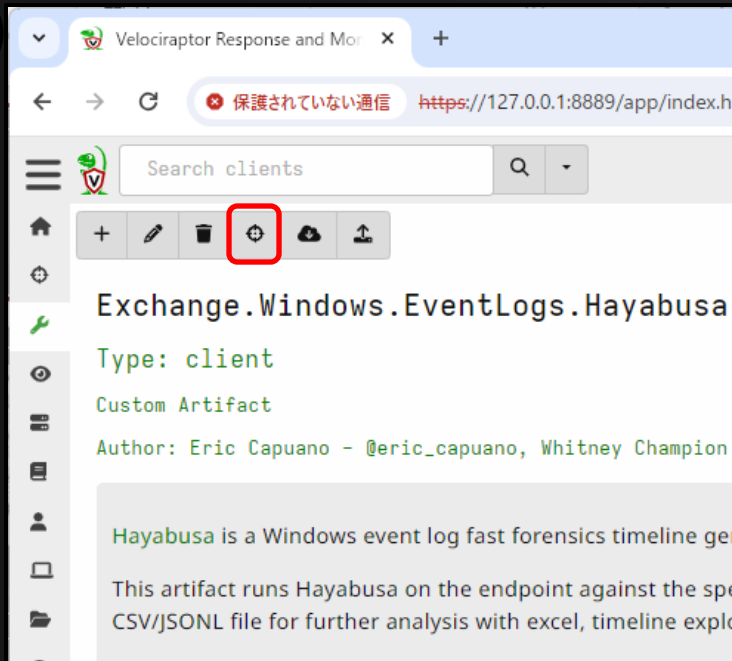
Yamato Security's automatic analysis



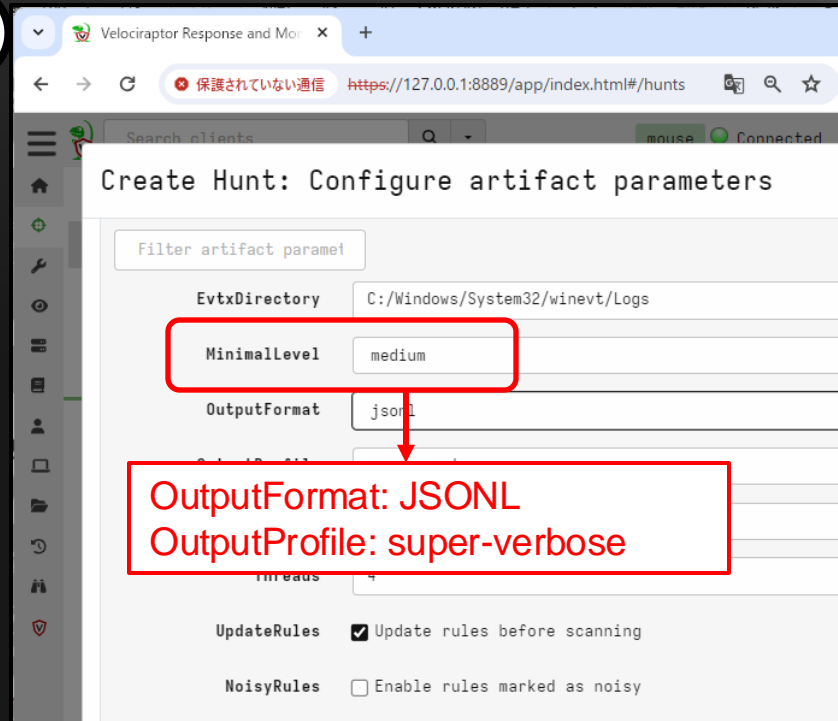
Collect Hayabusa timelines from multiple client at once



1



2



Collect Hayabusa timelines from multiple client at once



The screenshot displays the HITCON web interface. At the top, there's a search bar with "all" and a "win" status indicator showing "Connected". Below this is a toolbar with various icons. A table lists active hunts, with one entry for "H.CNVS70G6KTBS4" with the description "hayabusa". Below the table are pagination controls showing "Showing 1 to 1 of 1" and a "Goto Page" button. The main content area has tabs for "Overview", "Requests", "Clients", and "Notebook". The "Overview" tab is selected, showing details for the hunt: Artifact Names (Exchange.Windows.EventLogs.Hayabusa), Hunt ID (H.CNVS70G6KTBS4), Creator (admin), Creation Time (2024-03-24T06:10:42Z), Expiry Time (2024-03-31T06:09:57Z), State (RUNNING), Ops/Sec (Unlimited), CPU Limit (Unlimited), and IOPS Limit (Unlimited). To the right, the "Results" section shows "Total scheduled" (3) and "Finished clients" (3). Below this, there's a "Download Results" button and a list of "Available Downloads" including "H.CNVS70G6KTBS4-summary".

State	HuntId	Description	Created	Started	Expires
⌵	H.CNVS70G6KTBS4	hayabusa	2024-03-24T06:10:42Z	2024-03-24T06:10:42Z	2024-03-31T06:09:57Z

10 25 30 50 Showing 1 to 1 of 1 « 0 » Goto Page

Overview Requests Clients Notebook

Overview

Artifact Names Exchange.Windows.EventLogs.Hayabusa

Hunt ID H.CNVS70G6KTBS4

Creator admin

Creation Time 2024-03-24T06:10:42Z

Expiry Time 2024-03-31T06:09:57Z

State RUNNING

Ops/Sec Unlimited

CPU Limit Unlimited

IOPS Limit Unlimited

Results

Total scheduled 3

Finished clients 3

Download Results

Available Downloads

H.CNVS70G6KTBS4-summary

Uncompressed 132 Mb

Compressed 6 Mb

Hayabusa results for multiple clients can be downloaded in one file!

Takajo automagic command!



'automagic' command !

= Batch analysis of multiple files !

Takajo's various analysis commands and
Analyze multiple files in one go!

All analyzed by Takajo automagic!



Batch analysis only requires the following two steps!

1. **Download hunt JSONL results from Velociraptor GUI**
2. `takajo.exe automagic -t results.jsonl`

All analyzed by Takajo automagic!



```
PS C:\tmp\takajo-2.4.0-win> .\takajo automagic -t .\timeline.jsonl -q
Started the automagic command
```

Automatically executes as many commands as possible and output results to a new folder.

```
File: .\timeline.jsonl (45.27 MB)
Counting total lines. Please wait.
Total lines: 34,940
```

Scanning the Hayabusa timeline. Please wait.

```
100%|████████████████████| 34940/34940 [ 3.0s< 0.0s, 61.55k/sec]
```

Command	Results	Saved Files
extract-scriptblocks	PowerShell logs: 108	case-1/scriptblock-logs/Summary.csv (19.92 KB) case-1/scriptblock-logs/*.txt
list-domains	Domains: 0	case-1/ListDomains.txt (0 Bytes)
list-domains(detailed)	Domains: 2	case-1/ListDomainsDetailed.txt (16 Bytes)
list-hashes	MD5: 1,038 SHA1: 976 SHA256: 978 Import: 1,038	case-1/ListHashes-MD5.txt (4.88 KB) case-1/ListHashes-SHA1.txt (5.66 KB) case-1/ListHashes-SHA256.txt (9.02 KB) case-1/ListHashes-ImportHashes.txt (4.28 KB)
list-ip-addresses	IP addresses: 0	case-1/ListIP-Addresses.txt (0 Bytes)
stack-cmdlines	Unique cmdlines: 1,411	case-1/StackCmdlines.csv (814.13 KB)
stack-computers	Unique computers: 54	case-1/StackTargetComputers.csv (49.51 KB)
stack-computers	Unique computers: 3,571	case-1/StackSourceComputers.csv (383.32 KB)

Visualization of MITRE ATT&CK TTPs with automagic results



ATT&CK® Navigator

https://mitre-attack.github.io/attack-navigator/

Hayabusa detection result heatmap

Reconnaissance 10 techniques	Resource Development 8 techniques	Initial Access 10 techniques	Execution 14 techniques	Persistence 20 techniques	Privilege Escalation 14 techniques	Defense Evasion 43 techniques	Credential Access 17 techniques
Active Scanning (0/3)	Acquire Access	Content Injection	Cloud Administration Command	Account Manipulation (0/6)	Abuse Elevation Control Mechanism (1/5)	Abuse Elevation Control Mechanism (1/5)	Adversary-in-the-Middle (1/3)
Gather Victim Host Information (0/4)	Acquire Infrastructure (0/8)	Drive-by Compromise	Command and Scripting Interpreter (5/9)	BITS Jobs	Access Token Manipulation (3/5)	Access Token Manipulation (3/5)	Brute Force (1/4)
Gather Victim Identity Information (0/3)	Compromise Accounts (0/3)	Exploit Public-Facing Application	Container Administration Command	Boot or Logon Autostart Execution (5/14)	Account Manipulation (0/6)	BITS Jobs	Credentials from Password Stores (2/6)
Gather Victim Network Information (1/6)	Compromise Infrastructure (0/7)	External Remote Services	Deploy Container	Boot or Logon Initialization Scripts (1/5)	Boot or Logon Autostart Execution (5/14)	Build Image on Host	Exploitation for Credential Access
Gather Victim Org Information (0/4)	Develop Capabilities (1/4)	Hardware Additions	Exploitation for Client Execution	Browser Extensions	Boot or Logon Initialization Scripts (1/5)	Debugger Evasion	Forced Authentication
Phishing for Information (0/4)	Establish Accounts (0/3)	Phishing (1/4)	Inter-Process Communication (1/3)	Compromise Client Software Binary	Boot or Logon Initialization Scripts (1/5)	Deobfuscate/Decode Files or Information	Forge Web Credentials (0/2)
Search Closed Sources (0/2)	Obtain Capabilities (1/6)	Replication Through Removable Media	Native API	Create Account (2/3)	Create or Modify System Process (1/4)	Deploy Container	Input Capture (0/4)
Search Open Technical Databases (0/5)	Stage Capabilities (0/6)	Supply Chain	Scheduled Task/Job	Create or Modify System Process	Domain Policy	Direct Volume Access	Modify Authentication Process (1/8)

Stack analysis with automagic results



Windows PowerShell						
	Count	TgtUser	TgtComp	LogonType	SrcIP	SrcComp
1	40	takahashi	fs03vuln.offsec.lan	3 - NETWORK	10.23.123.11	
2	19	tanaka	rootdc1.offsec.lan	3 - NETWORK	10.23.23.9	-
3	14	sato	fs03vuln.offsec.lan	3 - NETWORK	10.23.23.9	
4	12	suzuki	fs02.offsec.lan	3 - NETWORK	10.23.23.9	-
5	13	ito	mssql01.offsec.lan	3 - NETWORK	10.23.23.9	-
6	12	watanabe	srvdefender01.offsec.lan	3 - NETWORK	10.23.123.11	-
7	11	yamamoto	rootdc1.offsec.lan	3 - NETWORK	10.23.23.9	-
8	10	shimizu	FS03.offsec.lan	3 - NETWORK	10.23.42.38	-
9	10	hayashi	fs01.offsec.lan	3 - NETWORK	10.23.23.9	-
10	10	saito	srvdefender01.offsec.lan	3 - NETWORK	10.23.42.22	-
11	1	qfasodiab	01566s-win16-ir.threebeesco.com	3 - NETWORK	172.16.66.142	04246W-WIN10

.\StackLogons.csv [Row 10/11, Col 1/6]

You can also analyze outliers that have different naming conventions than other users, suspicious...

All analyzed by Takajo automagic!



Many other analysis results can be output with a single run of automagic:

- List Domain/IP/Hash
- Stacking Scheduled Tasks
- Stacking Users
- Stacking Service names
- Task scheduler timeline
- Logon timeline
- Process timeline
- PowerShell execution history
- MITRE ATT&CK TTPs
- and more!

New command: html-report

- 2.6.0 "HITCON Community Release" new command!
- Graphical triage with HTML Report
- Raw data generated in SQLite file
- Timeline visualization for easy detection!

```
./takajo html-report -t ../hayabusa/timeline.jsonl -o html-report -r ../hayabusa/rules
```

```
Database file created.
```

```
Creating HTML report. Please wait.
```

```
HTML report completed.
```

```
Please open "../html-report/index.html"
```



Summary

> critical alerts (357)

> high alerts (200)

> med alerts (784)

> low alerts (26894)

Summary

Total detections

SEVERITY	NUMBER OF DETECTIONS	DETECTION RATE
critical	357	1.16%
high	200	0.65%
med	784	2.54%
low	26894	87.07%
info	2653	8.59%

Unique detections

SEVERITY	NUMBER OF DETECTIONS	DETECTION RATE
critical	3	3.85%
high	9	11.54%
med	23	29.49%
low	15	19.23%
info	28	35.90%

Dates with most total detections

SEVERITY	NUMBER OF DETECTIONS	DETECTION RATE
critical	2023-10-21	352



Summary

▼ critical alerts (357)

■Antivirus Exploitation Framework Detection (1)

mouse (1) (2023-11-04 ~ 2023-11-04)

■Antivirus Password Dumper Detection (4)

mouse (4) (2024-01-28 ~ 2024-08-02)

■Defender Alert (Severe) (352)

mouse (352) (2023-10-21 ~ 2024-08-04)

▼ high alerts (200)

■Antivirus Hacktool Detection (5)

mouse (5) (2023-11-04 ~ 2024-08-02)

■Antivirus Relevant File Paths Alerts (149)

mouse (149) (2023-10-22 ~ 2024-06-29)

■Defender Alert (High) (3)

mouse (3) (2024-01-28 ~ 2024-08-02)

■Important Log File Cleared (1)

DESKTOP-CNG7416 (1)

Summary

Total detections

SEVERITY	NUMBER OF DETECTIONS	DETECTION RATE
critical	357	1.16%
high	200	0.65%
med	784	2.54%
low	26894	87.07%
info	2653	8.59%

Unique detections

SEVERITY	NUMBER OF DETECTIONS	DETECTION RATE
critical	3	3.85%
high	9	11.54%
med	23	29.49%
low	15	19.23%
info	28	35.90%

Summary

[> critical alerts \(357\)](#)[> high alerts \(200\)](#)[> med alerts \(784\)](#)[> low alerts \(26894\)](#)

mouse

357

Critical

199

High

736

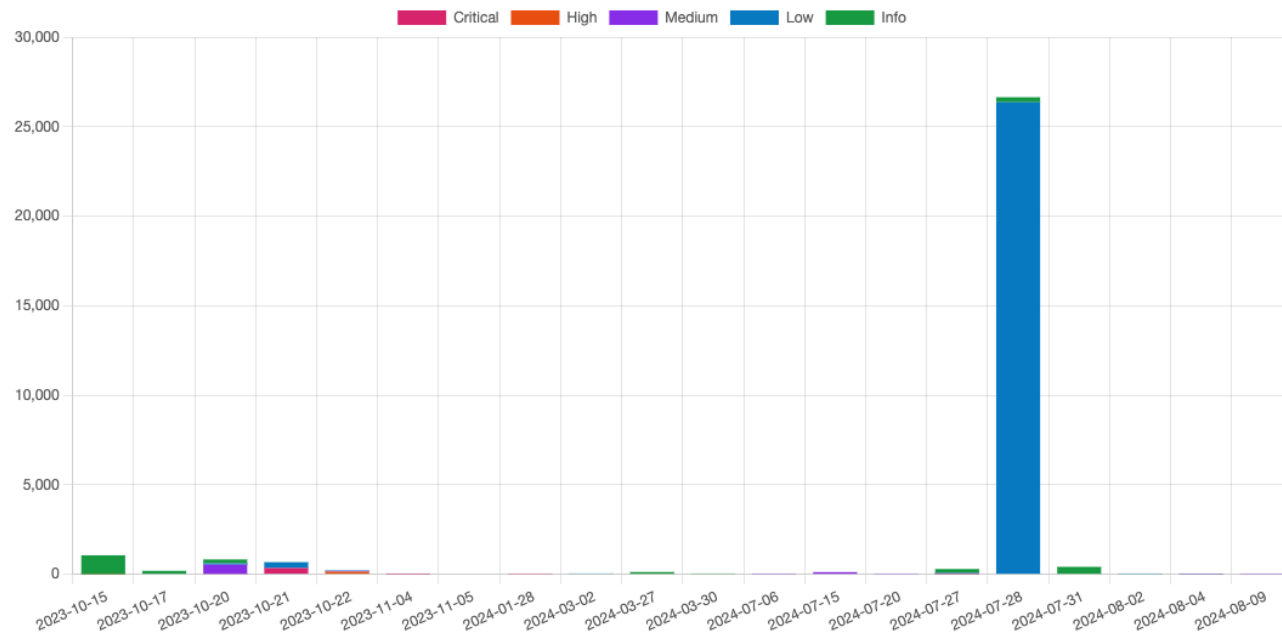
Medium

26893

Low

2589

Information



mouse

357

Critical

199

High

736

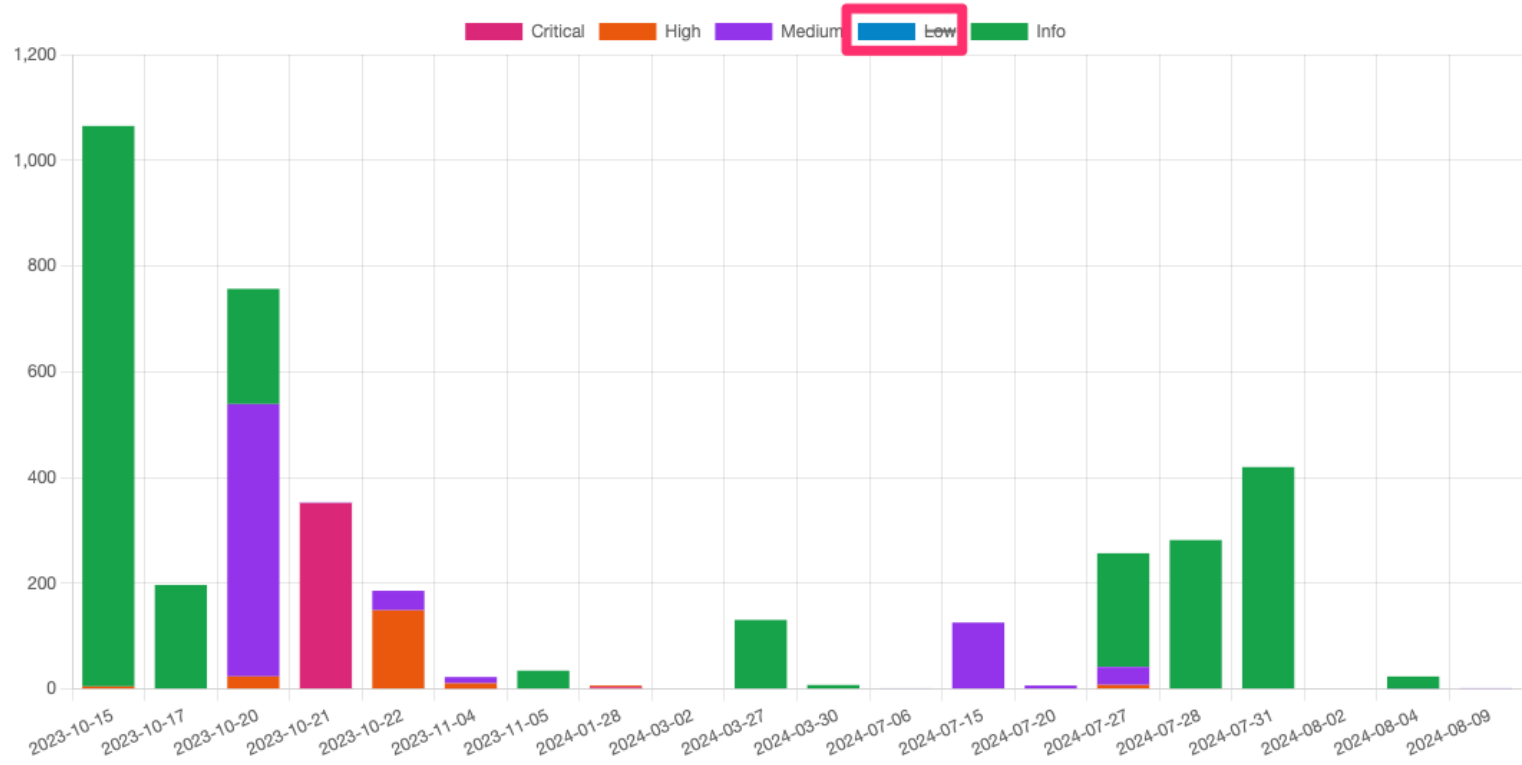
Medium

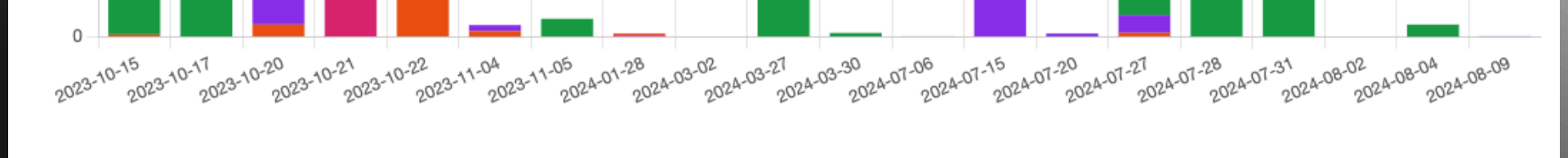
26893

Low

2589

Information





Detection Rule List

ALERT TITLE	RULE PATH	SEVERITY
Antivirus Exploitation Framework Detection	av_exploiting.yml	critical
Antivirus Password Dumper Detection	av_password_dumper.yml	critical
Defender Alert (Severe)	Defender_1116_Crit_Alert.yml	critical
Antivirus Hacktool Detection	av_hacktool.yml	high
Antivirus Relevant File Paths Alerts	av_relevant_files.yml	high
Defender Alert (High)	Defender_1116_High_Alert.yml	high
Microsoft Defender Blocked from Loading Unsigned DLL	win_security_mitigations_defender_load_unsigned_dll.yml	high
Microsoft Defender Tamper Protection Trigger	win_defender_tamper_protection_trigger.yml	high
Powershell Token Obfuscation - Powershell	posh_ps_token_obfuscation.yml	high

HTML Report Future Plans

- Create a dynamic web server
 - SQLite access will be done dynamically through an API
 - This will allow better scalability for large data
- We will still support HTML report output though
 - Just in case it is hard to share data through a web server

Open-source contribution and management advice

Open-source contribution advice

- If you are thinking of creating an open-source free tool or resource (guide, training, etc...), please do! Especially if you are blue team!
- There are too many offensive tools and guides and not enough blue team defensive resources
- Create something that will solve a problem you currently have for work.
 - Even if your tool or guide does not become very popular, you will still have helped yourself!
- If you don't know what to make, start off by contributing to other open-source projects. (Fixing bugs, adding features, etc...)
 - Learn how to use GitHub, create issues/pull requests
- It does not have to be a tool, it can be just documentation, but I recommend to manage on GitHub!
- It is hard work, but very well worth it!

Open-source management advice

- Make sure there is a leader you can trust or be that leader.
- Make your project and goals clear.
- Example: “I want to create a Windows event log analyzer that is flexible and supports Sigma rules. Let’s try to present at a conference!”
- Have a diversity of members with different strengths and weaknesses.
- One person usually isn’t a master at everything. You need people with good documentation skills, presentation skills, design skills, not just programming skills.
- Doing a project with 3-5 people is probably the best number.
- Have good documentation!!!
- As this is volunteer work, don’t force anyone to do anything, but keep people motivated and make sure there are no problems. If they need to focus on work or personal things, no problem. Let them do that and maybe one day they will come back to help out again. If not, no problem either.

Open-source management advice

- Never give up! Success does not usually come right way unless you are very lucky (or have good marketing!). Consistent improvement is key. It will take time for people to notice but if you have a good tool or resource then people will promote it for you and that is the best promotion! (Aim for “being so good that people cannot ignore you.”)
- Have periodic meetings over Zoom. (Once a week in the beginning. Once every 2-3 weeks later on.) Make the meetings fun too! Not just serious talk...
- Meet up in-person some day if you have never met offline before.
- If someone contributes to your project, be sure to thank them and offer them more issues to work on if they are interested.
- Always welcome more members to help out.
- Keep maintaining it! It is always sad to see good open-source projects get abandoned and become unusable...

Future plans

Future plans

- Analysis with AI
 - "Analyze the Hayabusa and Takajo results and write a 10-page forensics report" => "OK! No problem"
 - Challenges:
 - Needs to keep all of the data local
 - Needs to be accurate
 - Current AI does not produce long reports so we would first have to script things to ask many questions
- Analysis with machine learning (UEBA-type detection)
 - Example: Find any abnormal logins
- Suzaku: A Sigma-based event log analyzer for cloud logs (AWS, Azure, GCP)
- Please let us know if you want to help out with any of these!

Thank you so much for listening!



If you like our tools, please consider supporting us with a GitHub star!

Please give us feedback on what we can do better. Please also contact us if you want to help out.

Check out the latest release information on X-Twitter!



Follow:

[@SecurityYamato](https://twitter.com/SecurityYamato)

Thank you for your attention!