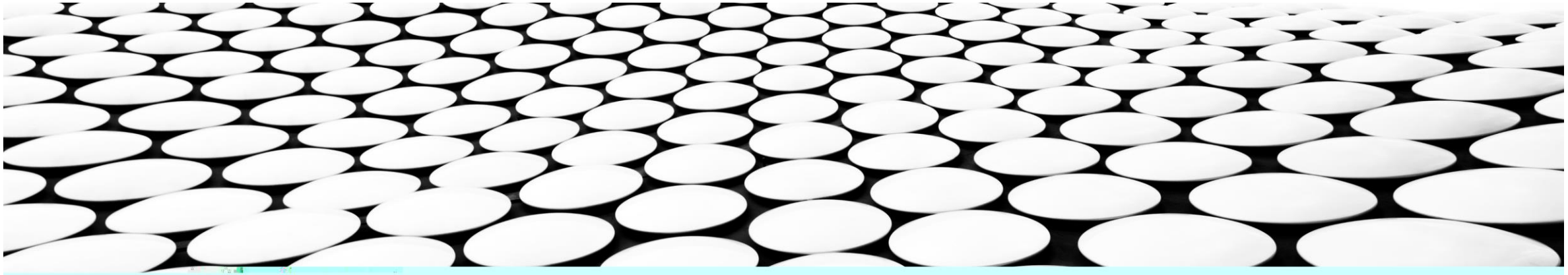
INVESTIGATING WINDOWS LOG WITH ELK

SIEM TACTICAL INTRODUCTION

THOMAS BILLAUT



AGENDA / OBJECTIVES

- My Needs and requirements
- What is a SIEM ?
- ELK to investigate
- SOF ELK distribution
- Analysing windows log

At the end of this workshop, I will be able to set up a SOF-ELK box to analyse windows logs in addition to native sof-elk capability (syslog, httpd, kape, passivedns, plaso)

NEEDS OR/AND REQUIREMENTS

I’VE A DREAM…

```
student@ubuntu:~/evtx/logs/logs_evtx$ file Security.evtx
Security.evtx: MS Windows Vista Event Log, 5642 chunks (no. 5641 in use), next record no. 506967, DIRTY
student@ubuntu:~/evtx/logs/logs_evtx$ head Security.evtx
ElfFile
Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event" System Provider Name="Microsoft-Windows-Security-Auditing" Guid="54849625-5478-4994-A5BA-3E3B0328C30D" EventSourceName="Security" />
Qualifiers
  Version
    Level
    Task
    Opcode
    Keywords
    TimeCreated SystemTime="
    EventRecordID
    Correlation
    Execution ProcessID="4" ThreadID="8" />
    Channel
    Computer
    Security />
</System>
<EventData>
<Data Name="SubjectUserSid">S-1-5-18</Data>
<Data Name="SubjectUserName">UBUNTU$</Data>
<Data Name="SubjectDomainName">WORKGROUP</Data>
<Data Name="SubjectLogonId">0x3e7</Data>
<Data Name="NewProcessId">0x1a4</Data>
<Data Name="NewProcessName">C:\Windows\System32\svchost.exe</Data>
<Data Name="TokenElevationType">%%1936</Data>
<Data Name="ProcessId">0x1a8</Data>
<Data Name="CommandLine">C:\Windows\System32\svchost.exe -k netsvcs -p -s Schedule</Data>
<Data Name="TargetUserSid">S-1-5-21-3623811015-3361044348-30300820-1013</Data>
<Data Name="TargetUserName">student</Data>
<Data Name="TargetDomainName">UBUNTU</Data>
<Data Name="TargetLogonId">0x2a4d9</Data>
<Data Name="ParentProcessName">C:\Windows\System32\services.exe</Data>
<Data Name="MandatoryLabel">S-1-16-16384</Data>
</EventData>
```

- Analysing easily amount of logs
- More precisely the evtx (windows + Vista) logs
- My workstation log…
  - Total size : [redacted] Mo
  - Number of files : 117
  - The biggest file -> Security.evtx / [redacted]4 Mo
  - How many fields ? More than 500

- Who has already tried to analyse windows log ?
  - Event viewer / eventvwr.msc?
  - Powershell ?
  - wevtutil.exe ?
  - EventLog Analyser ?
  - *SolarWinds® Log Analyzer…Euh?*

WHAT IS A SIEM

IF YOU ARE INTERESTED IN, GO TO SEC 555 ! WORSE IT !

- A tool, a technology :
  - Collection a variety of events
  - Establish context
  - Implement capabilities for real time analysis
  - Implement capabilities for historical analysis
  - Consist of : collectors, aggregator, broker, storage, search engine/report, alert
- Technology is not the limiting factor
- Lack of people, processes and expertise are key deficiencies
- Our job
  - Maximize the value of technology purchases
  - Planning (Requires knowledge of the enterprise) / thinking
    - Critical to decide data-collection strategy
    - Exercise discrimination in data we gathered
  - Running and building

“HAVING THE RIGHT PEOPLE CAN OFTEN HAVE THE MOST PROFOUND IMPACT ON THE OVERALL CAPABILITY OF & SOC"

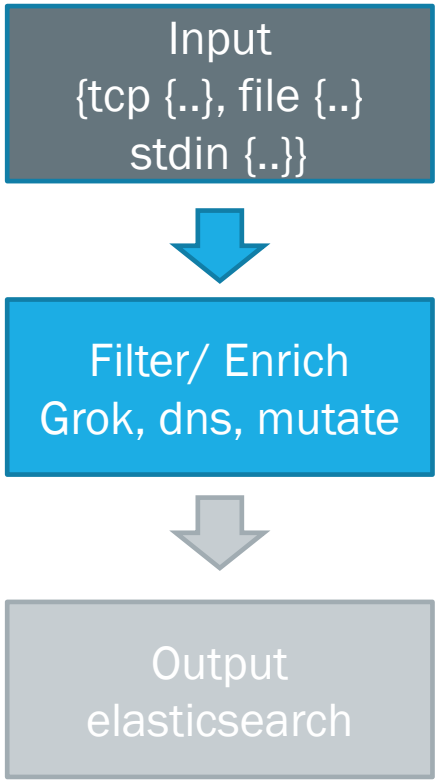
STATE OF SECURITY OPERATIONS WITH 2016 REPORT OF CAPABILITIES AND MATURITY OF CYBER DEFENSE ORGANIZATIONS BUSINESS WHITE PAPER

HTTPS://SSL.WWW8.HP.COM/US/EN/SSL/LEADGEN/SECURE_DOCUMENT.HTML?OBJID=4AA6-3593ENW&SIEBELID=560013401

ELK TO INVESTIGATE

A FREE SIEM…NOT REALLY FREE BUT CAN HELP EVERY ANALYST TO HAVE IS OWN SIEM

- Log collectors : agent, file, script…
- Log aggregator : logstash
- Broker (if needed) : kafka, rabbitmq
- Storage : Elasticsearch is a worm-based Storage platform
- Search : kibana
- Alert / automatise : Elastalert



SOF ELK DISTRIB

- SOF-ELK® is a “big data analytics” platform focused on the typical needs of computer forensic investigators/analysts and information security operations personnel.

•The platform is a customized build of the open source Elastic stack, consisting of the Elasticsearch storage and search engine, Logstash ingest and enrichment system, Kibana dashboard frontend, and Elastic Beats log shipper (specifically filebeat).

• /logstash/syslog/: Syslog-formatted data
•/logstash/nfarch/: Archived NetFlow output, formatted as described below
•/logstash/httpd/: Apache logs in common, combined, or vhost-combined formats
•/logstash/passivedns/: Logs from the passivedns utility
•/logstash/kape/: JSON-format files generated by the KAPE triage collection tool. (See this document for details on which specific output files are currently supported and their required file naming structure.)
•/logstash/plaso/: CSV bodyfile-format files generated by the Plaso tool from the log2timeline framework. (See this document for details on creating CSV files in a supported format.)

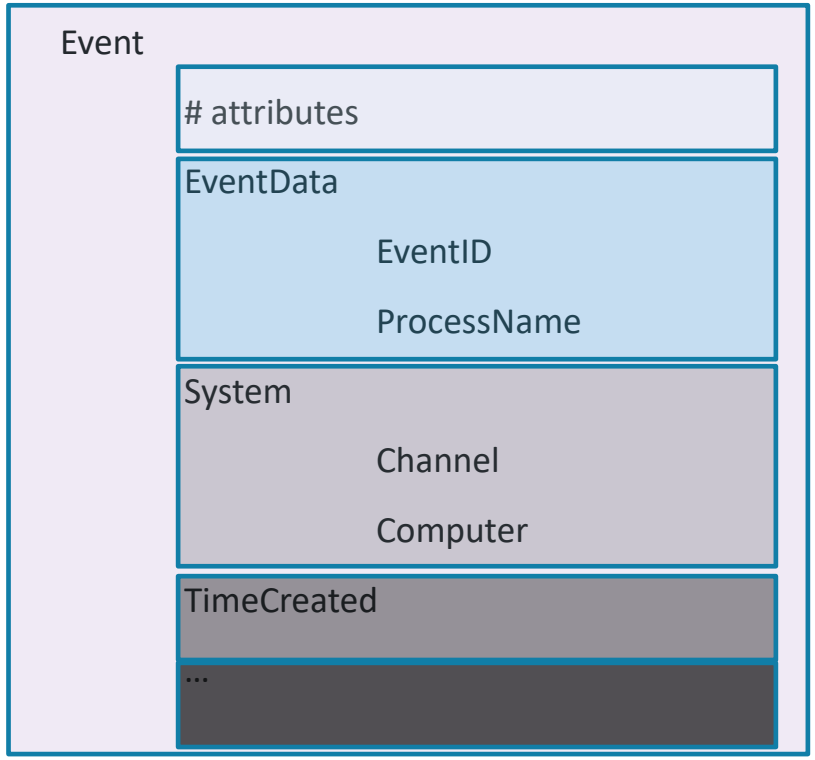
- With a significant amount of customization and ongoing development, SOF-ELK® users can avoid the typically long and involved setup process the Elastic stack requires.
- Instead, they can simply download the pre-built and ready-to-use SOF-ELK® virtual appliance that consumes various source data types (numerous log types as well as NetFlow), parsing out the most critical data and visualizing it on several stock dashboards.
- Advanced users can build visualizations the suit their own investigative or operational requirements, optionally contributing those back to the primary code repository.

SOF ELK DISTRIB

- Default credentials :
  - login : pass // elk_user : forensics
- Based on filebeat to pass the log to logstash
- Require only scp to drop the logs
- Cerebro (for deleting data in elasticseach already available)
- But can make it with netcat (to install : yum install nmap)
- Not ready for windows / Require to do custom configurations :
  - create a conf for filebeat windows.yml in path: /usr/local/sof-elk/lib/filebeat_inputs/*.yml
  - Modify the 1001-preprocess-json.conf
  - Modify the 6300-windows.conf (=> timestamp, source_ip, …)

SOF ELK DISTRIB

- EVTX LOG fomat



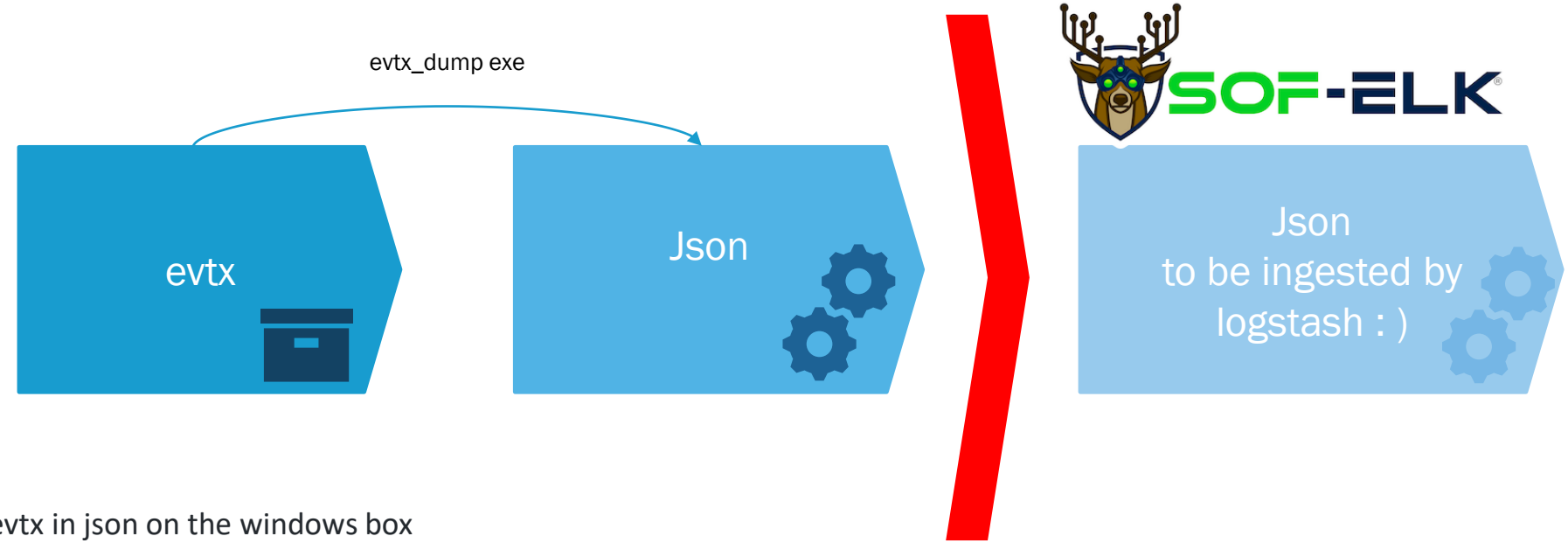
- My dilemma concerning the evtx to json transform

- Keep the original structure ? => OPTION 1
  - Event.EvenData.EventID for EventID
  - Sigma rules almost compatible (pseexec for ex) :
    - (winlog.event_data.Image.keyword:*\\PSEXESVC.exe AND winlog.event_data.User:"NT\ AUTHORITY\\SYSTEM")

- Reformat, direct, accessible value => OPTION 2
  - EventID, source_ip to facilitate logstash enrichment (tags, geo,…) and search
    - if [EventID] in [529, 530, 531, 532, 533, 534, 535, 536, 537, 538, 539, 4625, 4771] {
      mutate {
      add_tag => ["logon_failure", "alert_data"]
      }

SOF ELK DISTRIB

OPTION 1



- Transform evtx in json on the windows box
  TIPS for a loop : for /r %i in (*.evtx) do evtx_dump.exe -o jsonl -f .\evtx_to_json\%~ni.json %i
- Push the logs for elk ingestion on sof-elk
  TIPS : scp .\evtx_to_json* elk_user@192.168.xx.xx:/logstash/windows_json/
- Require specific configuration file : filebeat, logstash / 6302-windows-jsonl.conf
- PRO : quicker
- CONS :
  poorer (almost no enrichment by default)
  not working with some logs where Event.System.EventID is not existing…

Evtx_dump.exe :
https://github.com/omerbenamram/evtx/releases

SOF ELK DISTRIB

OPTION 1

- Filebeat conf (under root)
  - mkdir /logstash/windows_json
  - Chmod +777 /logstash/windows_json
  - Chmod +t /logstash/windows_json
  - Under /usr/local/sof-elk/lib/filebeat_inputs/ make windows_json.yml

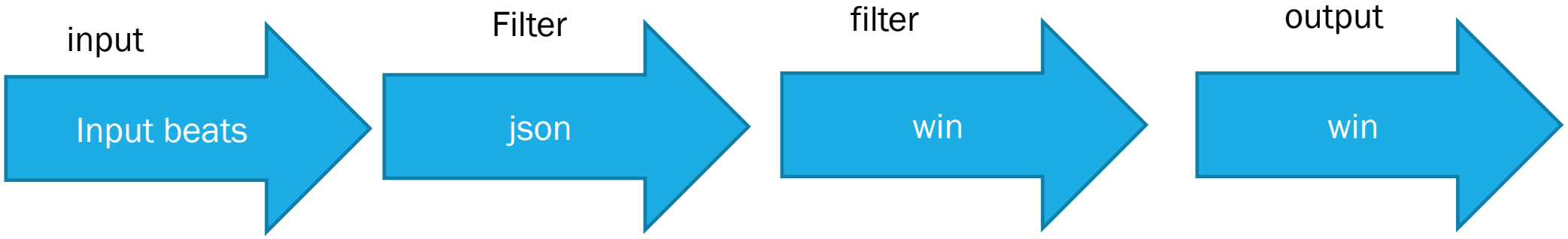
Windows_json.yml

```
# This file creates a filebeat prospector for windows json not normarlized evtx -> jsonlines log source data from the SOF-ELK® VM itself
- type: log
  paths:
    - /logstash/windows_json/*/*/*/*/*
    - /logstash/windows_json/*/*/*/*
    - /logstash/windows_json/*/*/*
    - /logstash/windows_json/*/*
    - /logstash/windows_json/*
  exclude_files: [ 'readme.txt', '\.gz$', '\.bz2$', '\.zip$' ]
  close_inactive: 5m
  fields_under_root: true
  fields:
    type: win
    tags : json
```

SOF ELK DISTRIB

OPTION 1

- Create windows logstash pipeline



0000-input-beats.conf
(existing)

1001-preprocess-json.conf (existing)
if "json" in [tags]

New file
6305_win.conf
If [type] = "win"
- @timestamp
- channel -> for indexing in elk (no uppercase, no space, ...)
- Eventid for processing

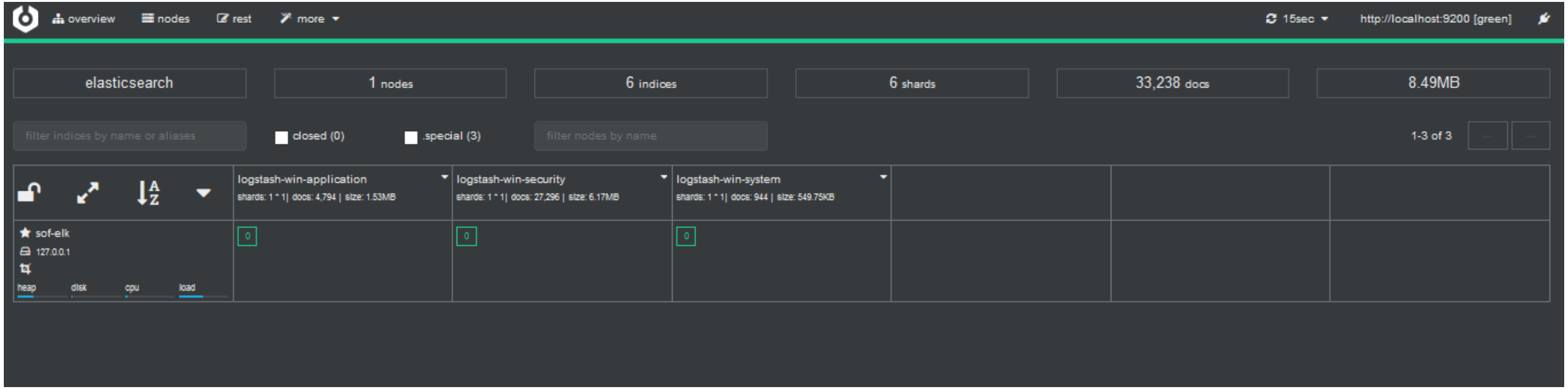
New file
9305-output-win.conf

Elastic custom Index

- Create link form /etc/logstash/conf.d
  - sudo ln -s /usr/local/sof-elk/configfiles/6305-win.conf /etc/logstash/conf.d/6305-win.conf
  - sudo ln -s /usr/local/sof-elk/configfiles/9305-output-win.conf /etc/logstash/conf.d/9305-output-win.conf
- Check :
  - sudo systemctl status logstash.service –l
  - cat /var/log/logstash/logstash-plain.log
- Go : scp security.json elk_user@192.168.xx.xx:/logstash/windows_json
- Check on cereabro http://192.168.xx.xx:9000
- Index on kibana http://192.168.xx.xx:5601

SOF ELK DISTRIB

OPTION 1



SOF ELK DISTRIB

OPTION 1

Management / Index patterns / logstash-win-security

Elasticsearch

- Index Management
- Index Lifecycle Policies
- Rollup Jobs
- Transforms
- Remote Clusters
- Snapshot and Restore
- License Management
- 8.0 Upgrade Assistant

Kibana

- Index Patterns
- Saved Objects
- Spaces
- Reporting
- Advanced Settings

logstash-win-security

Time Filter field name: @timestamp

This page lists every field in the **logstash-win-security** index and the field's associated core type as recorded by Elasticsearch. To change a field type, use the Elasticsearch Mapping API

Fields (332) | Scripted fields (0) | Source filters (0)

Filter | All field types

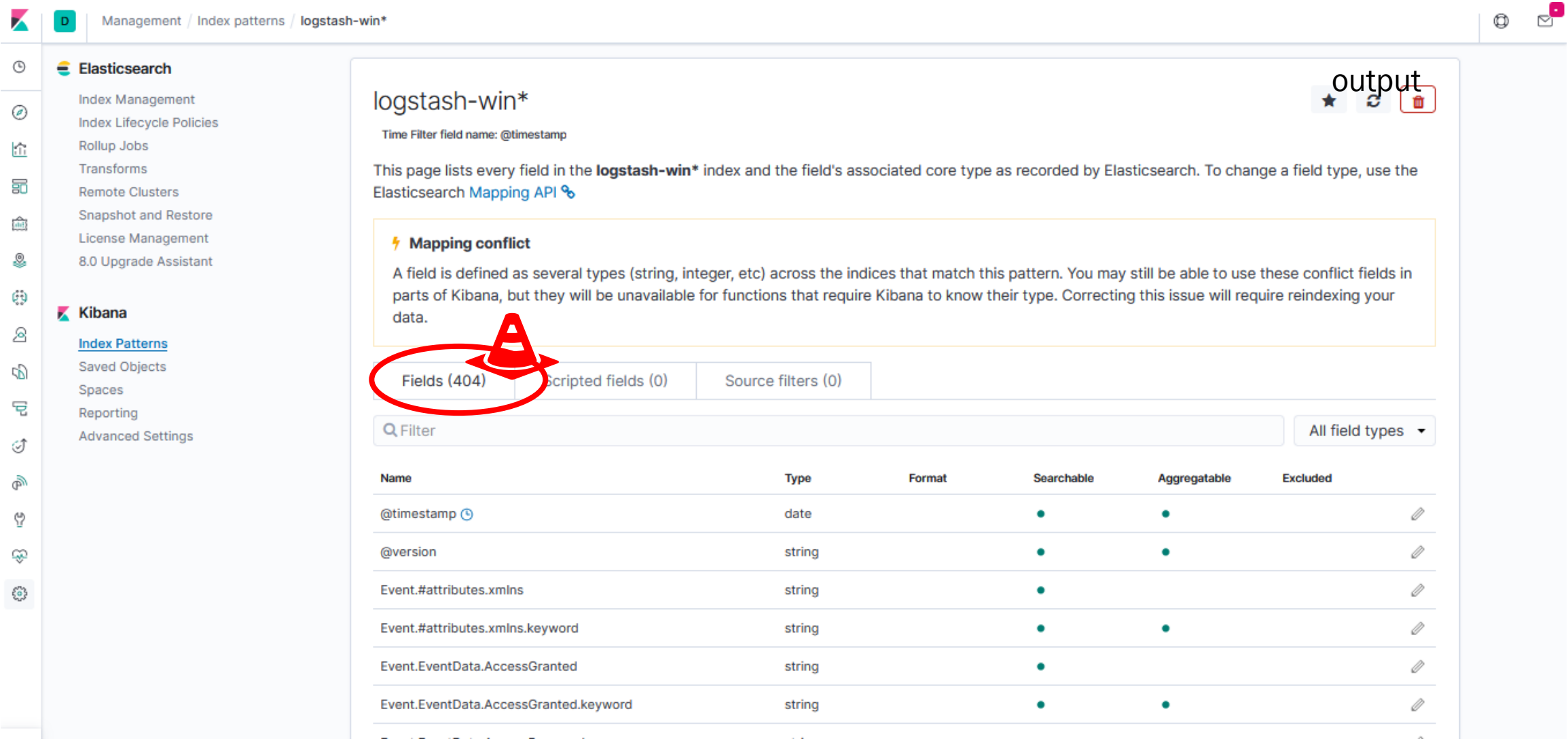
Name	Type	Format	Searchable	Aggregatable	Excluded
@timestamp	date		•	•	
@version	string		•	•	
Event.#attributes.xmlns	string		•		
Event.#attributes.xmlns.keyword	string		•	•	
Event.EventData.AccessGranted	string		•		
Event.EventData.AccessGranted.keyword	string		•	•	
Event.EventData.AccessRemoved	string		•		
Event.EventData.AccessRemoved.keyword	string		•	•	
Event.EventData.AccountExpires	string		•		
Event.EventData.AccountExpires.keyword	string		•	•	

Rows per page: 10

1 2 3 4 5 ... 34

SOF ELK DISTRIB

OPTION 1



SOF ELK DISTRIB

OPTION 1

Discover

New Save Open Share Inspect

Search KQL Last 90 days Show dates Refresh

+ Add filter

logstash-win-security

Event.EventData.

Filter by type 0

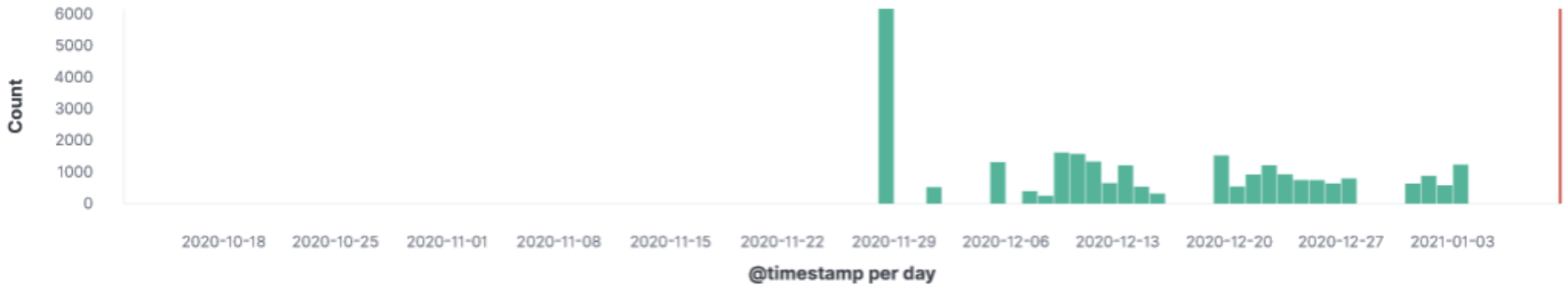
Selected fields

Available fields

- t Event.EventData.Aut...
- t Event.EventData.Co...
- t Event.EventData.Ele...
- t Event.EventData.Imp...
- t Event.EventData.IpA...
- t Event.EventData.IpP...
- # Event.EventData.Key...
- t Event.EventData.Lm...
- t Event.EventData.Log...

27,296 hits

2020-10-11 17:39:16.022 +00:00 - 2021-01-09 17:39:16.022 +00:00 — Auto



Time	_source
2021-01-03 17:21:57.192 +00:00	agent.type: filebeat agent.version: 7.6.1 agent.id: ef726016-50b2-4622-916a-a77ed798d5d1 agent.ephemeral_id: 08f8f7cb-91fe-4b76-a552-17e3a6b2aaad agent.hostname: sof-elk Event.EventData.TargetUserName: - Event.EventData.NewProcessId: 0x764 Event.EventData.ParentProcessName: C:\Windows\System32\svchost.exe Event.EventData.SubjectUserName: [redacted] [redacted]Event.EventData.TargetLogonId: 0x0 Event.EventData.ProcessId: 0x374 Event.EventData.CommandLine:
2021-01-03 17:21:57.155 +00:00	agent.type: filebeat agent.version: 7.6.1 agent.id: ef726016-50b2-4622-916a-a77ed798d5d1 agent.ephemeral_id: 08f8f7cb-91fe-4b76-a552-17e3a6b2aaad agent.hostname: sof-elk

SOF ELK DISTRIB

OPTION 1

Discover

New Save Open Share

EventID:4624 KQL Last 90 days Show dates Refresh

+ Add filter

logstash-win*

Search field names

Filter by type 0

Selected fields

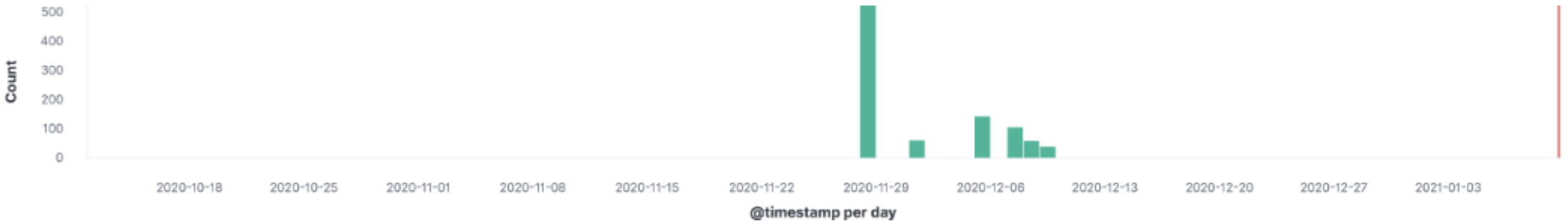
_source

Available fields

@timestamp
@version
Event.#attributes.xmlns
Event.EventData.AccountName
Event.EventData.Authenticatio...
Event.EventData.ElevatedToken
Event.EventData.ImagePath
Event.EventData.Impersonatio...
Event.EventData.IpAddress
Event.EventData.IpPort
Event.EventData.KeyLength

925 hits

2020-10-11 18:10:09.073 +00:00 - 2021-01-09 18:10:09.073 +00:00 — Auto



Time	_source
2020-12-09 09:06:06.123 +00:00	agent.type: filebeat agent.version: 7.6.1 agent.id: ef726016-50b2-4622-916a-a77ed798d5d1 agent.ephemeral_id: 08f8f7cb-91fe-4b76-a552-17e3a6b2aaad agent.hostname: sof-elk Event.EventData.ProcessName: C:\Windows\System32\services.exe Event.EventData.TargetOutboundDomainName: - Event.EventData.TargetLinkedLogonId: 0x0 Event.EventData.SubjectUserName [redacted] Event.EventData.TransmittedServices: - Event.EventData.TargetLogonId: 0x3e7 Event.EventData.TargetUserSid: S-1-5-18 Event.EventData.IpPort: - Event.EventData.ElevatedToken: %%1842 Event.EventData.SubjectDomainName: WORKGROUP Event.EventData.RestrictedAdminMode: - Event.EventData.KeyLength: 0
2020-12-09 09:05:53.390 +00:00	agent.type: filebeat agent.version: 7.6.1 agent.id: ef726016-50b2-4622-916a-a77ed798d5d1 agent.ephemeral_id: 08f8f7cb-91fe-4b76-a552-17e3a6b2aaad agent.hostname: sof-elk Event.EventData.ProcessName: C:\Windows\System32\services.exe Event.EventData.TargetOutboundDomainName: - Event.EventData.TargetLinkedLogonId: 0x0 Event.EventData.SubjectUserName [redacted] Event.EventData.TransmittedServices: - Event.EventData.TargetLogonId: 0x3e7 Event.EventData.TargetUserSid: S-1-5-18 Event.EventData.IpPort: - Event.EventData.ElevatedToken: %%1842 Event.EventData.SubjectDomainName: WORKGROUP Event.EventData.RestrictedAdminMode: - Event.EventData.KeyLength: 0
2020-12-09 09:05:53.350 +00:00	agent.type: filebeat agent.version: 7.6.1 agent.id: ef726016-50b2-4622-916a-a77ed798d5d1 agent.ephemeral_id: 08f8f7cb-91fe-4b76-a552-17e3a6b2aaad

SOF ELK DISTRIB

OPTION 1

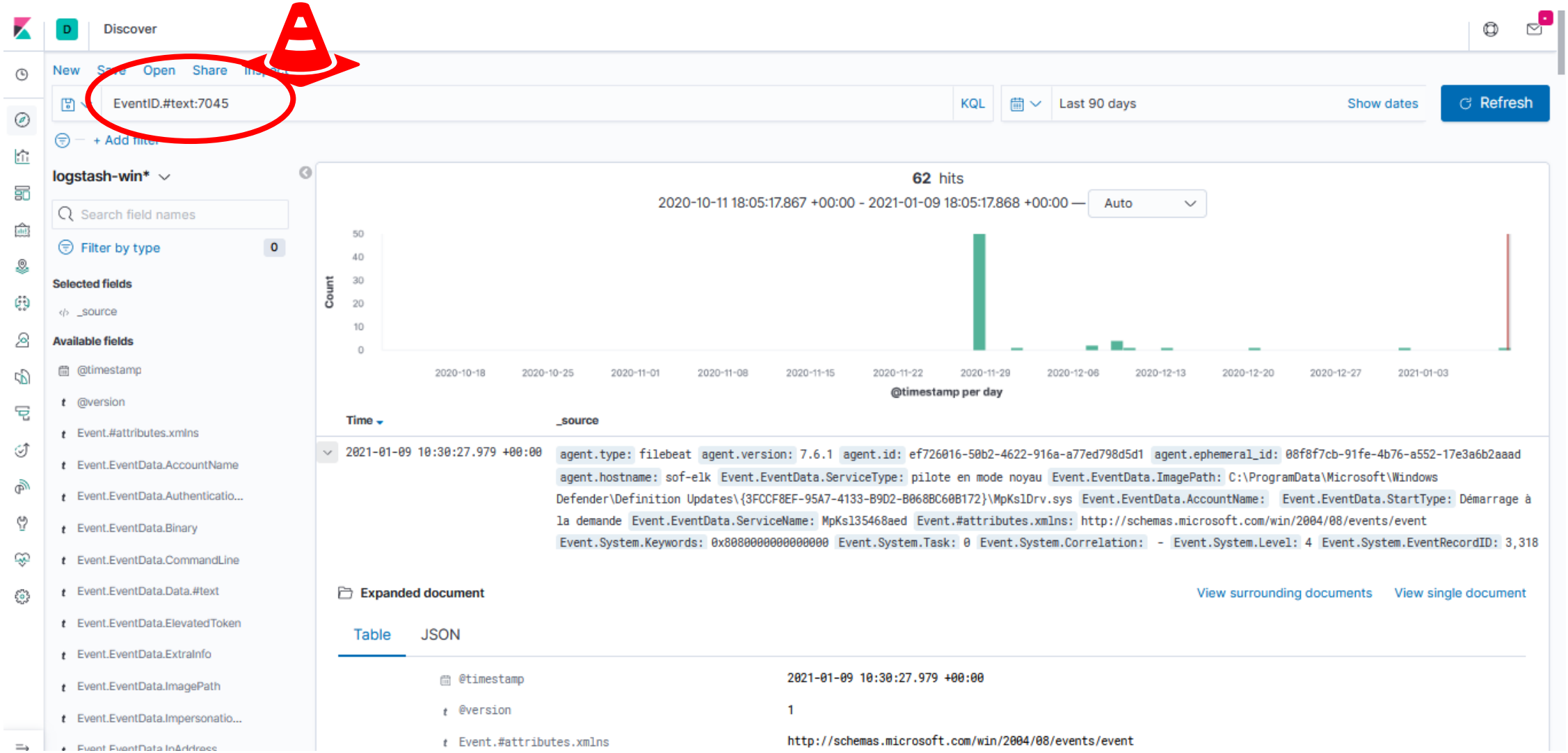
Discover

- Event.EventData.TargetOutbou...
- Event.EventData.TargetOutbou...
- Event.EventData.TargetUserNa...
- Event.EventData.TargetUserSid
- Event.EventData.TokenElevatio...
- Event.EventData.TransmittedS...
- Event.EventData.VirtualAccount
- Event.EventData.Workstation
- Event.EventData.WorkstationN...
- Event.System.Correlation
- Event.System.Correlation.#attri...
- Event.System.EventRecordID
- Event.System.Execution.#attri...
- Event.System.Execution.#attri...
- Event.System.Keywords
- Event.System.Level
- Event.System.Opcode
- Event.System.Provider.#attribu...
- Event.System.Provider.#attribu...
- Event.System.Security
- Event.System.Task
- Event.System.Version

Event.EventData.TargetLinkedLogonId	0x0
Event.EventData.TargetLogonId	0x3e7
Event.EventData.TargetOutboundDomainName	-
Event.EventData.TargetOutboundUserName	-
Event.EventData.TargetUserName	Système
Event.EventData.TargetUserSid	S-1-5-18
Event.EventData.TransmittedServices	-
Event.EventData.VirtualAccount	%%1843
Event.EventData.WorkstationName	-
Event.System.Correlation.#attributes.ActivityID	84E41319-CC6B-0001-3113-E4846BCCD601
Event.System.EventRecordID	9,786
Event.System.Execution.#attributes.ProcessID	716
Event.System.Execution.#attributes.ThreadID	11,008
Event.System.Keywords	0x8020000000000000
Event.System.Level	0
Event.System.Opcode	0
Event.System.Provider.#attributes.Guid	54849625-5478-4994-A5BA-3E3B0328C30D
Event.System.Provider.#attributes.Name	Microsoft-Windows-Security-Auditing
Event.System.Security	-
Event.System.Task	12,544
Event.System.Version	2
EventID	4,624

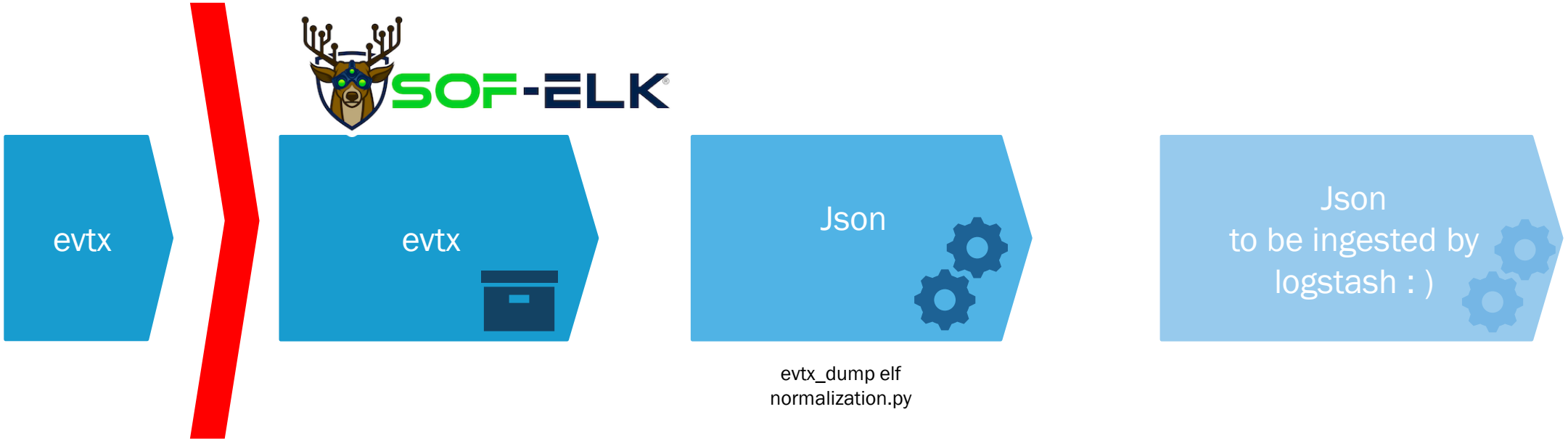
SOF ELK DISTRIB

OPTION 1



SOF ELK DISTRIB

OPTION 2



- Push the evtx logs on sof-elk in /tmp/windows/evtx

  scp c:\windows\system32\winevt\logs* elk_user@192.168.xx.xx:/tmp/windows/evtx

- Run windows_transform.sh script on the linux box :

  bash +x windows_transform.sh

  - Transform evtx in json in /tmp/windows/json
  - Normalize file from /tmp/windows/json to a filebeat directory /logstash/windows

- Requirements : windows_transform.sh , elf (evtx_dump), a python script (normalize.py) and filebeat specific configuration file
- PRO : most effective, enrichment, easier search (across all type of logs for EventID for example)
- CONS : longer than option 1

SOF ELK DISTRIB

OPTION 2

- Filebeat conf (under root)
  - mkdir /logstash/windows
  - Chmod +777 /logstash/windows
  - Chmod +t /logstash/windows
  - Under /usr/local/sof-elk/lib/filebeat_inputs/ make windows.yml

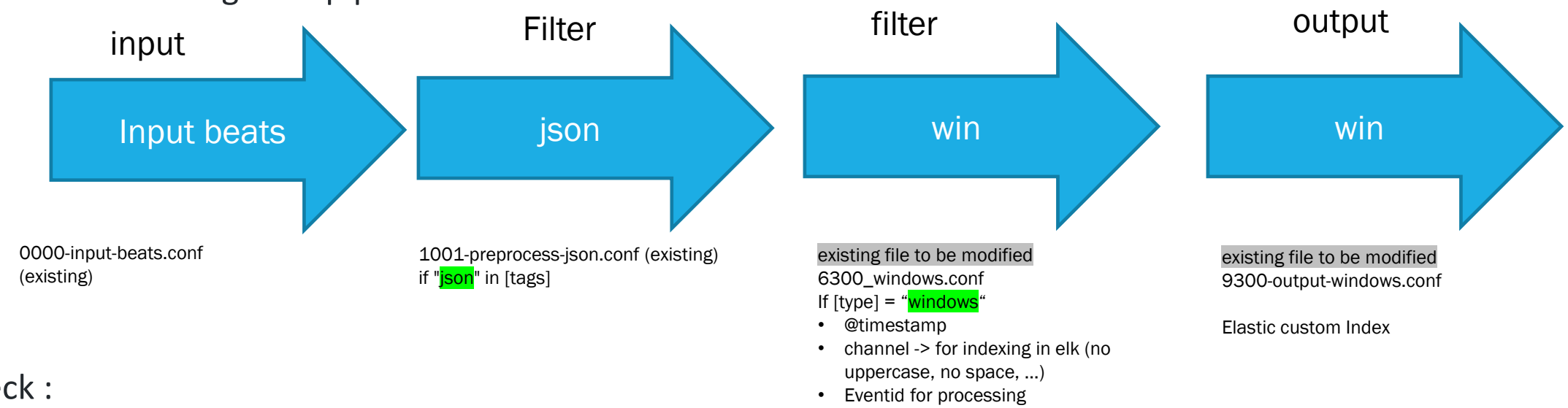
Windows.yml

```
# This file creates a filebeat prospector for windows json normarlized log source data from the SOF-ELK® VM itself
- type: log
  paths:
    - /logstash/windows/*/*/*/*/*
    - /logstash/windows/*/*/*/*
    - /logstash/windows/*/*/*
    - /logstash/windows/*/*
    - /logstash/windows/*
  exclude_files: [ 'readme.txt', '\.gz$', '\.bz2$', '\.zip$' ]
  close_inactive: 5m
  fields_under_root: true
  fields:
    type: windows
    tags : json
```

SOF ELK DISTRIB

OPTION 2

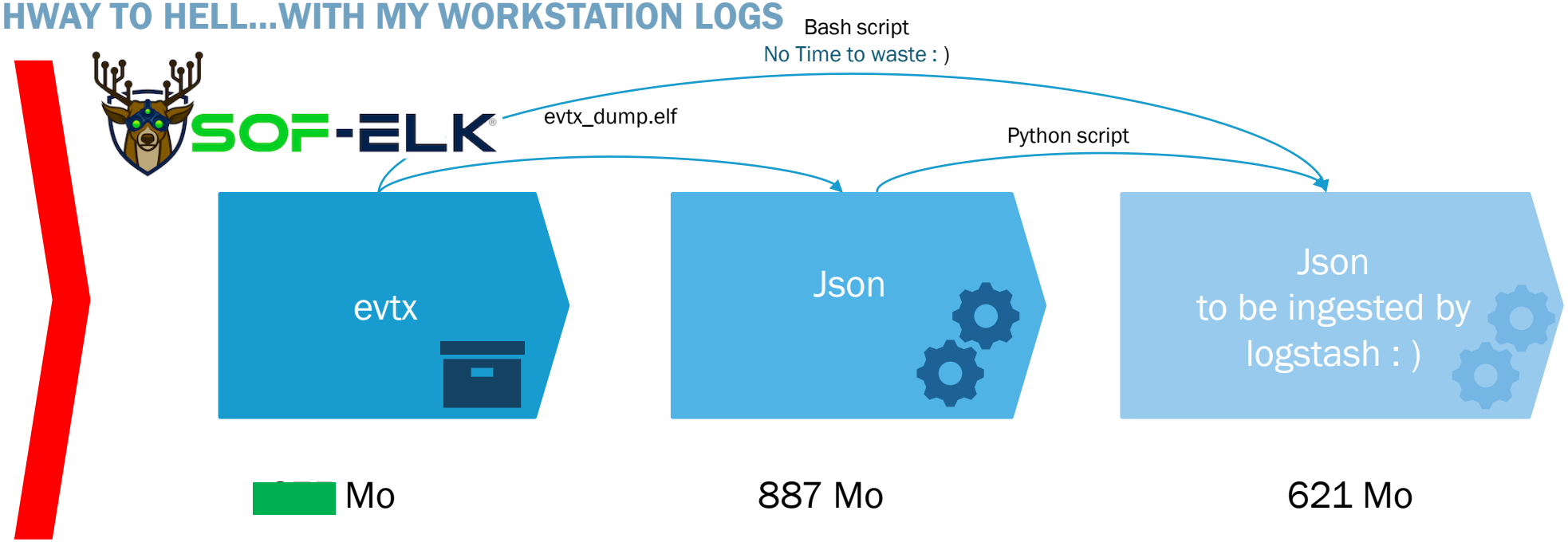
- Create windows logstash pipeline



- Check :
  - sudo systemctl status logstash.service –l
  - cat /var/log/logstash/logstash-plain.log
- Go : scp security.json elk_user@192.168.xx.xx:/logstash/windows_json
- Check on cereabro http://192.168.xx.xx:9000
- Index on kibana http://192.168.xx.xx:5601

FROM EVTX TO CUSTOM JSON

HIGHWAY TO HELL…WITH MY WORKSTATION LOGS



0,17% loss / 1 696 logs lost

Only on Microsoft-Client-Licensing-Platform%4Admin.evtx

Evtx_dump.exe :
https://github.com/omerbenamram/evtx/releases

FROM EVTX TO CUSTOM JSON

EVTX FOR DUMMIES

Unfortunatelly all the logs are not normalized…Normalization headache…

- Lot of properties : EventID is an integer most of the cases, except when it’s a dictionnary
- Event is the main evtx property, but it can also be a single property of EventData…
- Lots of attributes
- UserData…
- And a single log loss can be the explanation you’re loosing forever…

BASELING EVTX

WORKING WITH JSON AND JQ

- jq - commandline JSON processor
- Interactive learning on jq, jqplay : https://jqplay.org/
- List all EventID in your log :
  for f in `ls`; do echo "[+] - list of event in $f"; cat $f | jq .EventID | sort | uniq | sort -n ; echo; done
- 1193 EventID…hopefully without being pawned ;)
- Looking for a specific event : cat <log_file>.json | jq '. | select(.'EventID'==1102)' …Boom !

BASELINING EVTX

BASELING EVTX

- Security (main resource but not exclusive)
  https://www.ultimatewindowssecurity.com/securitylog/encyclopedia/

4624 - login, 4625 - logout, 4648 – explicit cred, 4688 – process, 4720 – account created, 4722 – account enabled, 4724/4738 Additional user creation events, 4728 / addition to a security enabled global group, 4732 / addition to a security enabled global group, 4697 – service installed, 4698 – scheduled tack created (+4699, 4700/01/02/)

- System

7045 - A service was installed in the system

7030 - Service is marked as an interactive service. However, the system is configured to not allow interactive services. This service may not function properly.

1056 – Create RDP certificate

- Microsoft-Windows-AppLocker%4EXE_and_DLL_ 8001 / 8002 /8003 / 8004

8003 - (EXE/MSI) was allowed to run but would have been prevented from running if the AppLocker policy were enforced

8004 - (EXE/MSI) was prevented from running.

BASELINING EVTX

BASELING EVTX

- Powershell
  - Microsoft-Windows-PowerShell/Operational.evtx

    4103 et 4104 logging of all PowerShell command input and output

    Event ID 4104: Script Block Logging
  - Windows Powershell.evtx

- WinRM
  - Microsoft-WindowsWinRM/Operational.evtx

- RDP

https://ponderthebits.com/2018/02/windows-rdp-related-event-logs-identification-tracking-and-investigation/

  - Microsoft-Windows-Terminal-Services-RemoteConnectionManager/Operational

    EventID: 1149 / User authentication succeeded (= connection ok, bu
  - Microsoft-Windows-TerminalServices-LocalSessionManager/Operational /

    Eventid 21 / Remote Desktop Services: Session logon succeeded

    EventID: 22 / Remote Desktop Services: Shell start notification received

    EventID : 24 / Remote Desktop Services: Session has been disconnected

    EventID: 25 / Remote Desktop Services: Session reconnection succeeded

    EventID : 40 / Session <X> has been disconnected, reason code <Z>

BASELINING EVTX

BASELING EVTX

- Defender
  - Microsoft-Windows-WindowsDefender/Operational

1116 : Windows Defender has detected malware or other potentially unwanted software

1117: Windows Defender has taken action to protect this machine from malware or other potentially unwanted

- To go further

Événements à surveiller

https://docs.microsoft.com/fr-fr/windows-server/identity/ad-ds/plan/appendix-l--events-to-monitor

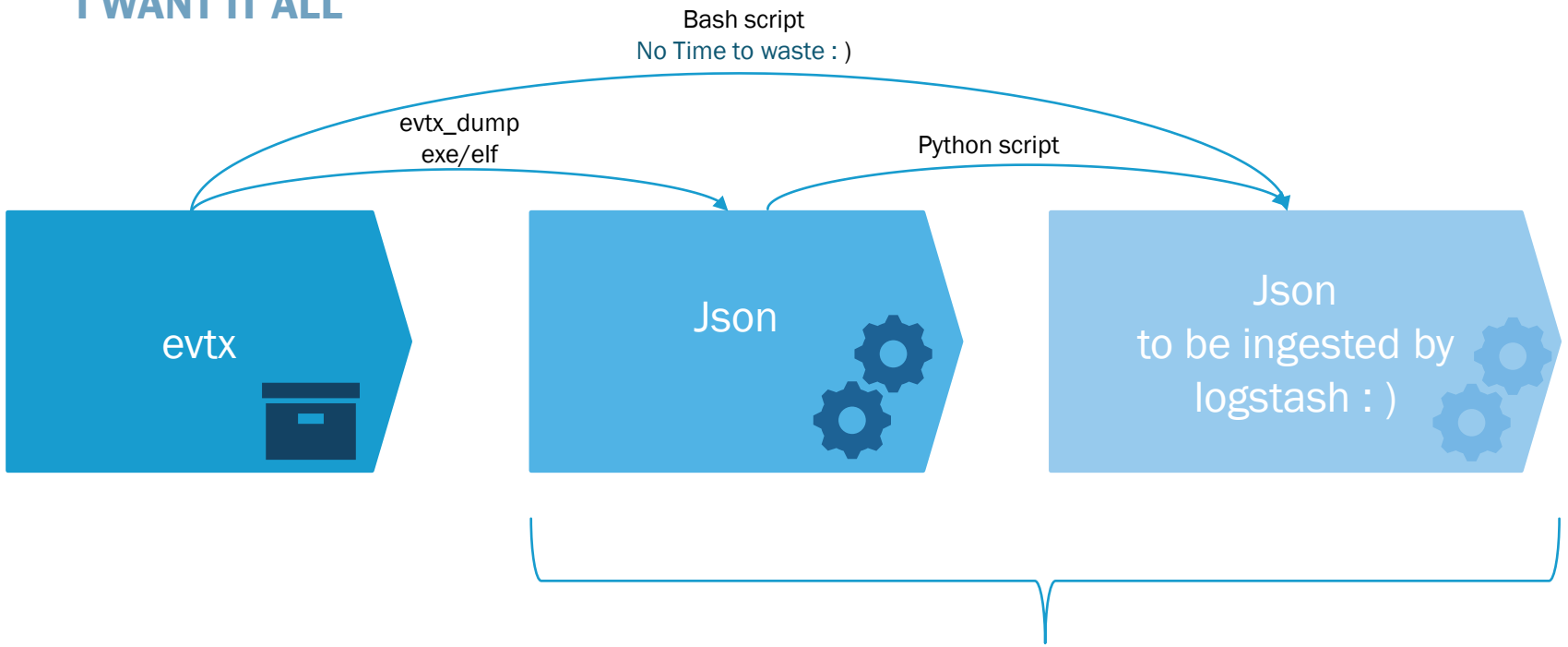
BASELINING EVTX

WHICH LOG SOURCES TO PRIORITIZE?

- Security
- System
- Application
- Firewall
- Microsoft-Windows-AppLocker%4EXE_and_DLL.
- Microsoft-Windows-Powershell*
- Microsoft-WindowsWinRM
- Microsoft-Windows-Terminal-Services*
- Sysmon if any

ANALYSING LOGS

I WANT IT ALL

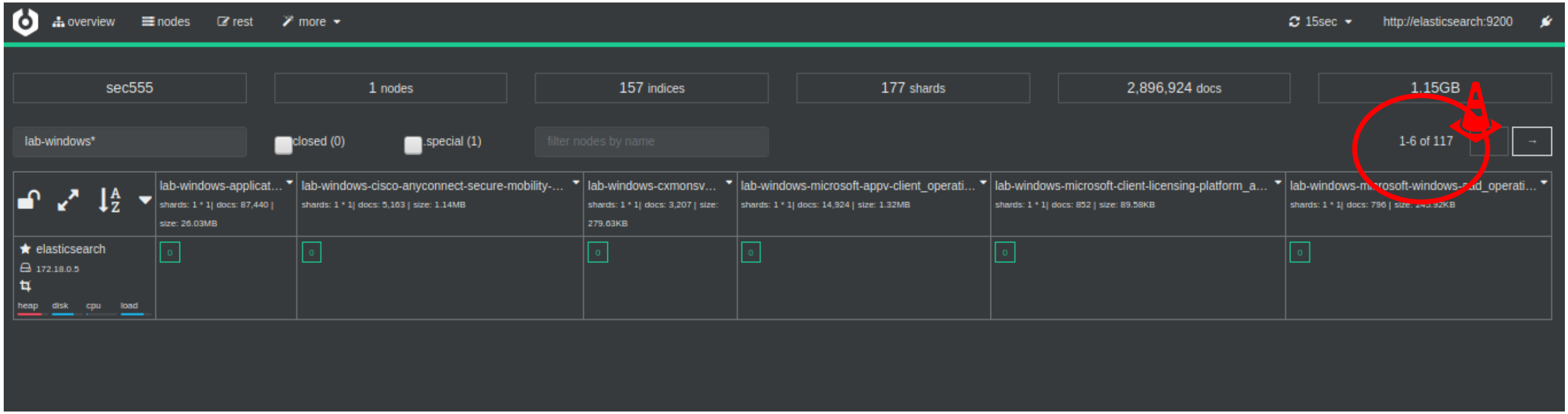


- One line json = raw text
- Grep inside :
  grep -P -o "\d{1,3}\.\d{1,3}\.\d{1,3}\.\d{1,3}[^\d]"
- Or jq if you re looking something specific

- The logstash conf is Everything
- Input, filter and output
- Input as stdin, file, socket
- Filter…
- Output elastic or broker

ANALYSING LOGS

I WANT IT ALL



ANALYSING LOGS

I WANT IT ALL BUT IT'S NOT POSSIBLE…NOT MORE 100 INDEXES

kibana

Discover
Visualize
Dashboard
Timelion
Dev Tools
Management

Management / Kibana

Index Patterns Saved Objects Advanced Settings

- lab1.4-complete*
- elastalert_status
- lab-bootcamp-day1
- lab-bootcamp-day3
- lab-dtf-bro-level3-dns
- lab-dtf-bro-level3-http
- lab-dtf-bro-level3-ssl
- lab-dtf-level3-windows
- lab-eburo*
- lab-eburo-security
- lab-enrich
- lab-win-application
- lab1.1-aggregator_only
- lab1.1-broker
- lab1.2
- lab1.2-complete
- lab1.3
- lab1.3-complete
- lab2.1-complete*
- lab2.2-complete
- lab3.1-complete
- lab3.2-complete
- lab3.3-complete
- lab4.1-complete
- lab4.2-complete
- lab4.3-complete

Create index pattern

Kibana uses index patterns to retrieve data from Elasticsearch indices for things like visualizations.

Include system indices

Step 1 of 2: Define index pattern

Index pattern

lab-windows

You can use a * as a wildcard in your index pattern.
You can't use empty spaces or the characters \, /, ?, ", <, >, |.

Next step

Your index pattern doesn't match any indices, but you have **100 indices** which look similar.

- lab-windows-application
- lab-windows-cisco-anyconnect-secure-mobility-client
- lab-windows-cxmonsvclog
- lab-windows-microsoft-appv-client_operational
- lab-windows-microsoft-client-licensing-platform_admin
- lab-windows-microsoft-windows-aad_operational
- lab-windows-microsoft-windows-appid_operational
- lab-windows-microsoft-windows-application-experience_program-compatibility-assistant
- lab-windows-microsoft-windows-application-experience_program-telemetry

ANALYSING LOGS

LEAN MANAGEMENT :)

```
student@ubuntu:~/evtx/logs/logs_logstash_json$ find ./ -size -300Ko
find: invalid -size type `o'
student@ubuntu:~/evtx/logs/logs_logstash_json$ find ./ -size -300k
./
./Microsoft-Windows-Application-Experience%4Program-Compatibility-Assistant_lgst.json
./Microsoft-Windows-Resource-Exhaustion-Detector%4Operational_lgst.json
./Microsoft-Windows-Diagnosis-Scheduled%4Operational_lgst.json
./Microsoft-Windows-WebAuthN%4Operational_lgst.json
./Microsoft-Windows-Containers-Wcifs%4Operational_lgst.json
./Microsoft-Windows-AppLocker%4EXE_and_DLL_lgst.json
./Microsoft-Windows-SmbClient%4Security_lgst.json
./Microsoft-Windows-Dhcpv6-Client%4Admin_lgst.json
./Microsoft-Windows-Ntfs%4WHC_lgst.json
./Microsoft-Windows-Policy%4Operational_lgst.json
./Microsoft-Windows-ReadyBoost%4Operational_lgst.json
./Microsoft-Windows-RemoteAssistance%4Operational_lgst.json
./Microsoft-Windows-Containers-Wcnfs%4Operational_lgst.json
./Microsoft-Client-Licensing-Platform%4Admin_lgst.json
./Microsoft-Windows-DeviceManagement-Enterprise-Diagnostics-Provider%4Admin_lgst.json
./Microsoft-Windows-WindowsSystemAssessmentTool%4Operational_lgst.json
./Microsoft-Windows-Windows_Firewall_With_Advanced_Security%4FirewallDiagnostics_lgst.json
./Microsoft-Windows-EapHost%4Operational_lgst.json
./Microsoft-Windows-DeviceSetupManager%4Operational_lgst.json
./microsoft-windows-diagnosis-scripted%4operational_lgst.json
./Microsoft-Windows-Storage-ClassPnP%4Operational_lgst.json
./Microsoft-Windows-EapMethods-RasTls%4Operational_lgst.json
student@ubuntu:~/evtx/logs/logs_logstash_json$ find ./ -size -300k | wc -l
23
```

ANALYSING LOGS

LEAN MANAGEMENT :)

From input driven (collect everything) to output driven (collect only what you know you need)

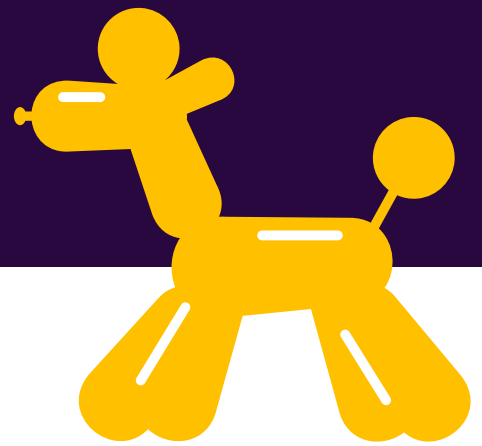
./Microsoft-Windows-Application-Experience%4Program-Compatibility-Assistant_lgst.json
./Microsoft-Windows-Resource-Exhaustion-Detector%4Operational_lgst.json -> 630 events (useless)
./Microsoft-Windows-Diagnosis-Scheduled%4Operational_lgst.json -> 238 events
./Microsoft-Windows-WebAuthN%4Operational_lgst.json
./Microsoft-Windows-Containers-Wcifs%4Operational_lgst.json
./Microsoft-Windows-AppLocker%4EXE_and_DLL_lgst.json
./Microsoft-Windows-SmbClient%4Security_lgst.json
./Microsoft-Windows-Dhcpv6-Client%4Admin_lgst.json => 0 IP but the v4 has some info ;)
./Microsoft-Windows-Ntfs%4WHC_lgst.json => only EventID 100 => no info => useless
./Microsoft-Windows-Policy%4Operational_lgst.json => 1 EventID => useless
./Microsoft-Windows-ReadyBoost%4Operational_lgst.json => 1 EventID => useless
./Microsoft-Windows-RemoteAssistance%4Operational_lgst.json => 2 EventID => useless
./Microsoft-Windows-Containers-Wcnfs%4Operational_lgst.json => 1 EventID => useless
./Microsoft-Client-Licensing-Platform%4Admin_lgst.json
./Microsoft-Windows-DeviceManagement-Enterprise-Diagnostics-Provider%4Admin_lgst.json
./Microsoft-Windows-WindowsSystemAssessmentTool%4Operational_lgst.json
./Microsoft-Windows-Windows_Firewall_With_Advanced_Security%4FirewallDiagnostics_lgst.json
./Microsoft-Windows-EapHost%4Operational_lgst.json
./Microsoft-Windows-DeviceSetupManager%4Operational_lgst.json
./microsoft-windows-diagnosis-scripted%4operational_lgst.json
./Microsoft-Windows-Storage-ClassPnP%4Operational_lgst.json
./Microsoft-Windows-EapMethods-RasTls%4Operational_lgst.json

Microsoft-Windows-Application-Experience%4Program-Compatibility-Assistant
Microsoft-Windows-Application-Experience%4Program-Telemetry
Microsoft-Windows-Resource-Exhaustion-Detector%4Operational
Microsoft-Windows-Diagnosis-Scheduled%4Operational_lgst.json
Microsoft-Windows-Diagnosis-DPS%4Operational_lgst.json
Microsoft-Windows-Diagnosis-PCW%4Operational_lgst.js
microsoft-windows-diagnosis-scripted%4operational_lgst.json
Microsoft-Windows-WebAuthN%4Operational_lgst.json
Microsoft-Windows-Containers-Wcifs%4Operational_lgst.json
Microsoft-Windows-Containers-Wcnfs%4Operational_lgst.json
Microsoft-Windows-Dhcpv6-Client%4Admin_lgst.json
Microsoft-Windows-Ntfs%4WHC_lgst.json
Microsoft-Windows-Policy%4Operational_lgst.json
Microsoft-Windows-ReadyBoost%4Operational_lgst.json
Microsoft-Windows-RemoteAssistance%4Operational_lgst.json
Microsoft-Client-Licensing-Platform%4Admin_lgst.json
Microsoft-Windows-DeviceManagement-Enterprise-Diagnostics-Provider%4Admin_lgst.jsc
Microsoft-Windows-DeviceSetupManager%4Operational_lgst.json
Microsoft-Windows-EapHost%4Operational_lgst.json
Microsoft-Windows-EapMethods-RasTls%4Operational_lgst.json
Microsoft-Windows-Diagnosis-PCW%4Operational_lgst.json
Pulse_Secure%4Operational_lgst.jso

ANALYSING LOGS

USELESS LOGS

```
student@ubuntu:~/evtx/logs/logs_logstash_json$ cat ./Microsoft-Windows-Application-Experience%4Program-Compatibility-Assistant_lgst.json | jq .EventID | sort | uniq
17
student@ubuntu:~/evtx/logs/logs_logstash_json$ head -n1 ./Microsoft-Windows-Application-Experience%4Program-Compatibility-Assistant_lgst.json | jq
{
  "Channel": "Microsoft-Windows-Application-Experience/Program-Compatibility-Assistant",
  "Computer": "W[redacted]r",
  "Correlation": null,
  "EventID": 17,
  "EventRecordID": 1,
  "Keywords": "0x4000000000000000",
  "Level": 4,
  "Opcode": 0,
  "Security": {
    "#attributes": {
      "UserID": "S-1-5-18"
    }
  },
  "Task": 0,
  "Version": 0,
  "SystemTime": "2020-03-06T18:10:11.060203Z"
}
```



ANALYSING LOGS

INTERESTING LOGS

Microsoft-Windows-Dhcp-Client%4Admin_lgst.json = networkHintString / point d’accès wifi, hwaddress

Microsoft-Windows-Ntfs%4Operational_lgst.json => volume name, guid, process name…

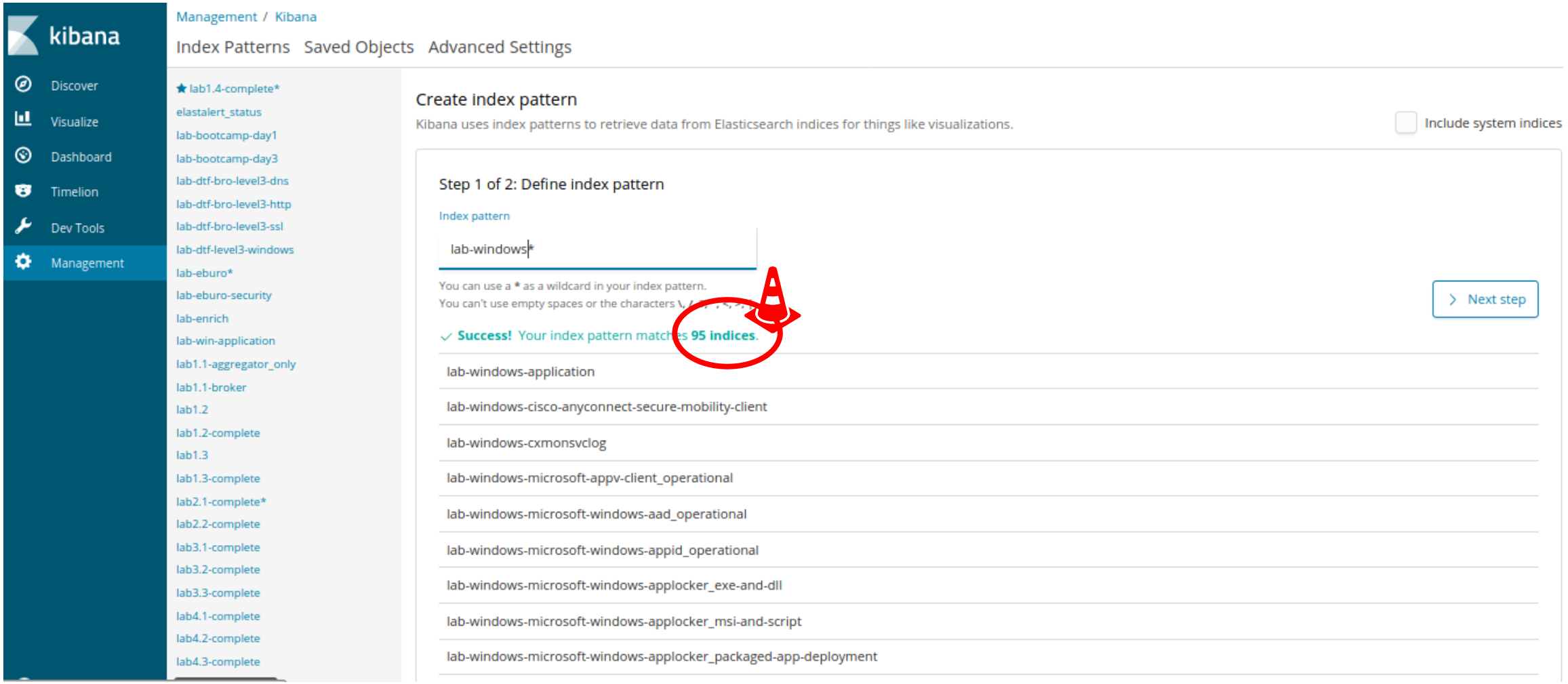
Microsoft-Windows-Storage-ClassPnP%4Operational_lgst.json => model of devoce connecterd

"IF YOU ARE MOVING FASTER THAN YOU DOCUMENT, YOU ARE GOING TOO FAST"

ANY EXPERIENCED ANALYST

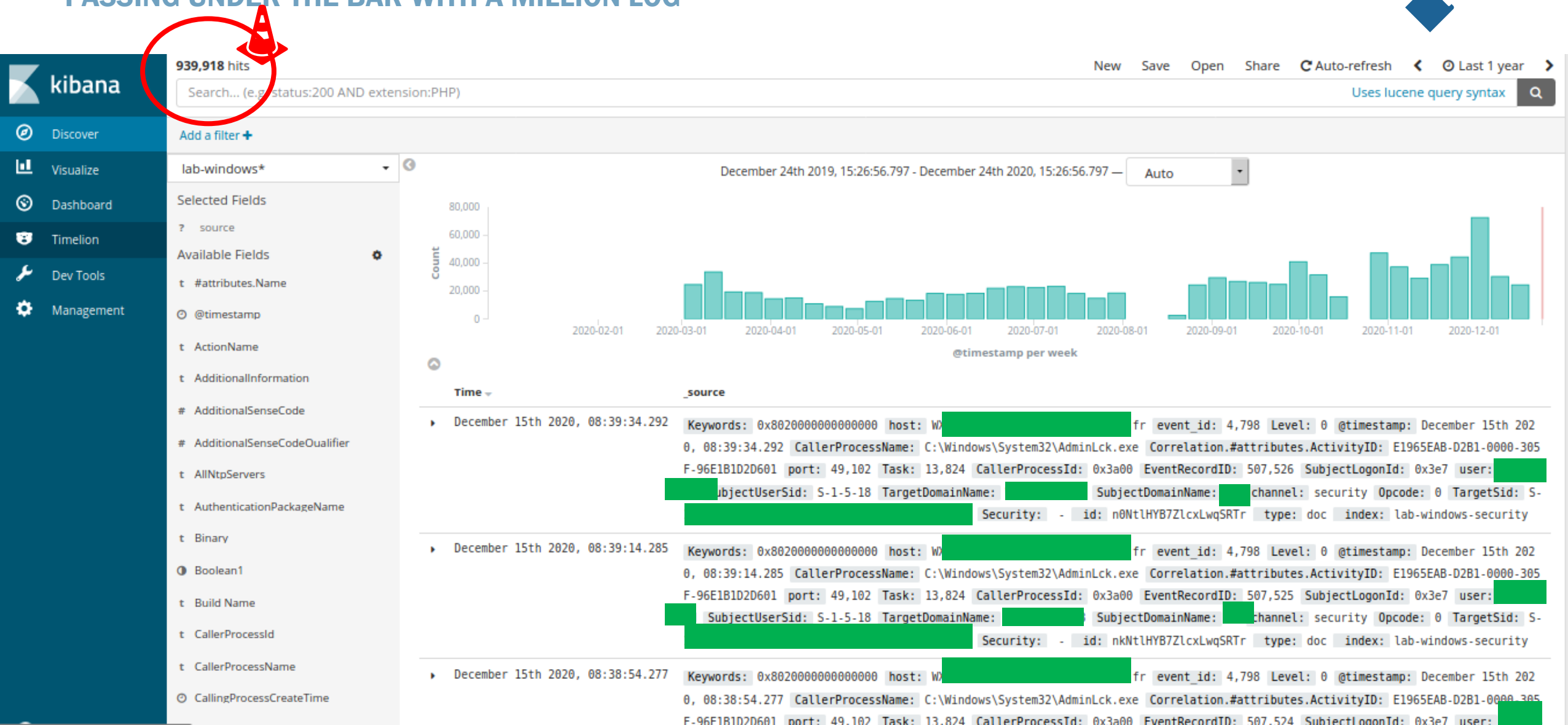
ANALYSING LOGS

PASSING UNDER THE BAR



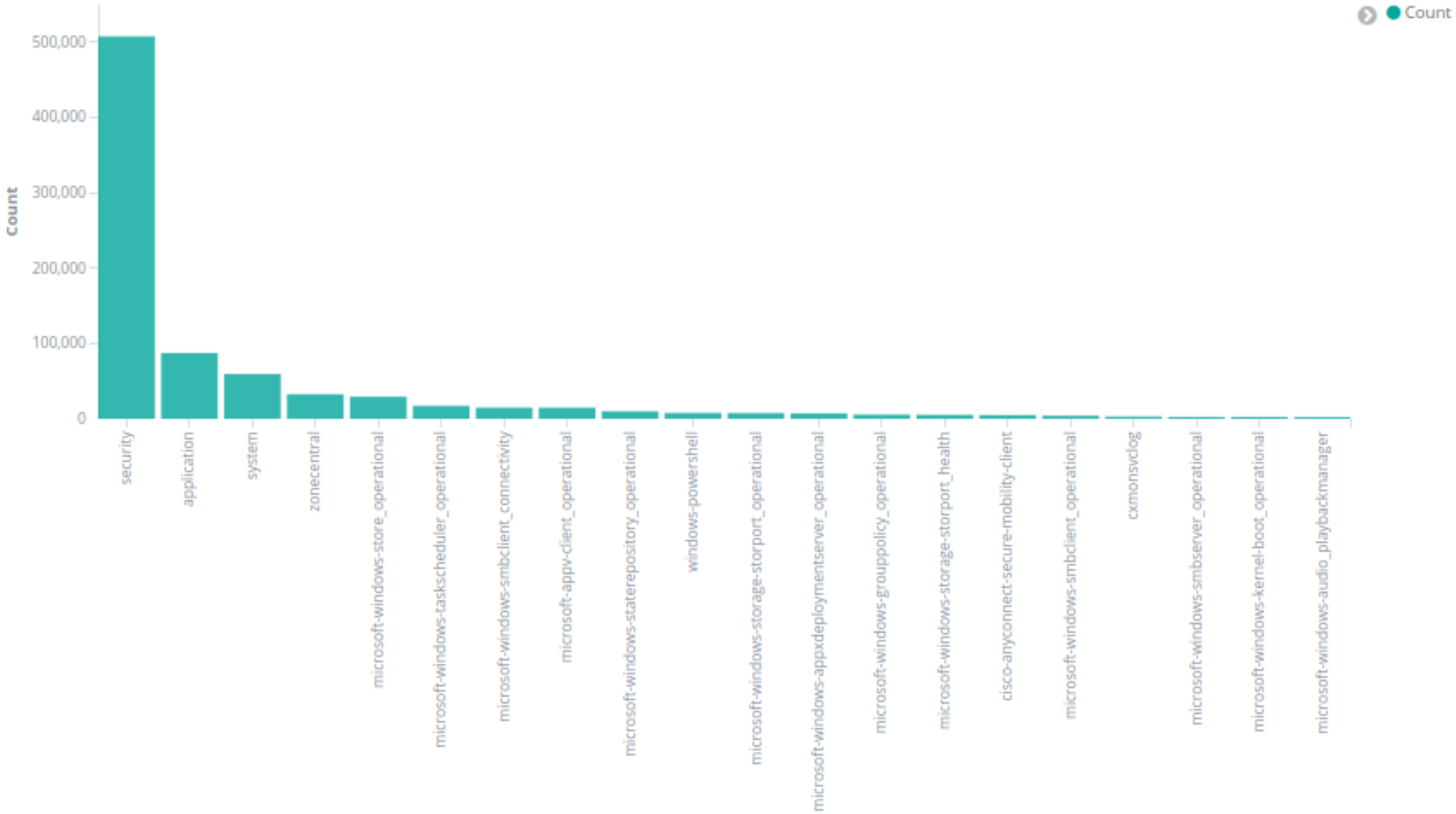
ANALYSING LOGS

PASSING UNDER THE BAR WITH A MILLION LOG



ANALYSING LOGS

WINDOWS CHANNEL DISTRIBUTION…MORE THAN 50% IS SECURITY ON EBURO WORKSTATION



ANALYSING LOGS

STARTING ANALYSIS..LET'S START WITH 4624...

ANALYSING LOGS

IF THERE'S ONLY ONE…LOGON TYPE 3…

kibana

Discover
Visualize
Dashboard
Timelion
Dev Tools
Management

1 hit

New Save Open Share Auto-refresh Last 1 year

user: [redacted] AND event_id:4624

Uses lucene query syntax

user: UPRI6463 AND event_id:4624

lab-windows*

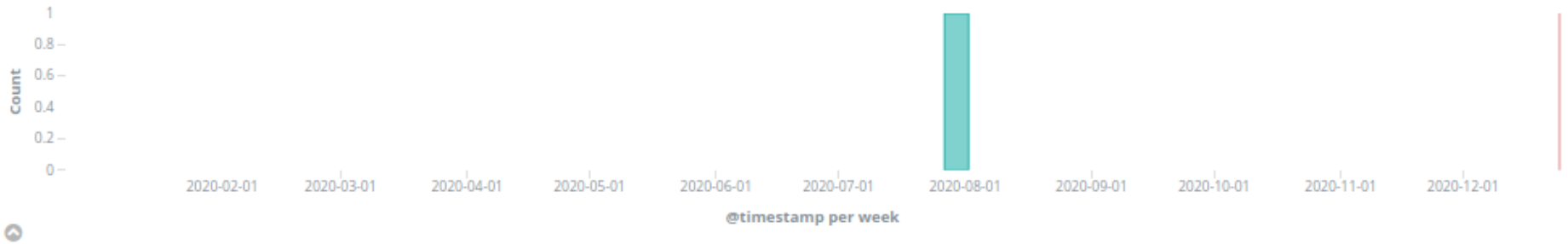
Selected Fields

? source

Available Fields

@timestamp
t AuthenticationPackageName
t Correlation.#attributes.Activi...
t ElevatedToken
EventRecordID
t ImpersonationLevel
t IpPort
KeyLength
t Keywords
Level
t LmPackageName
t LogonGuid

December 24th 2019, 15:47:10.628 - December 24th 2020, 15:47:10.629 — Auto



Time	_source
July 28th 2020, 16:24:39.742	user: [redacted] host: W[redacted]r event_id: 4,624 LmPackageName: - Level: 0 @timestamp: July 28th 2020, 16:24:39.742 Correlation.#attributes.ActivityID: FD3806C7-64AC-0002-1507-38FDAC64D601 port: 49,102 client_ip: [redacted] TargetLogonId: 0x3f02704 TargetOutboundUserName: - ProcessName: - SubjectLogonId: 0x0 VirtualAccount: %%1843 KeyLength: 0 TargetOutboundDomainName: - SubjectDomainName: - channel: security Opcode: 0 IpPort: 64166 LogonType: 3 ProcessId: 0x0 TargetLinkedLogonId: 0x0 Keywords: 0x8020000000000000 WorkstationName: - LogonGuid: A0CF5672-A28A-1FC1-FE67-F61B6775A3AF

ANALYSING LOGS

4624...AND EASY FILTER

kibana

Discover
Visualize
Dashboard
Timelion
Dev Tools
Management

Add a filter +

lab-windows*

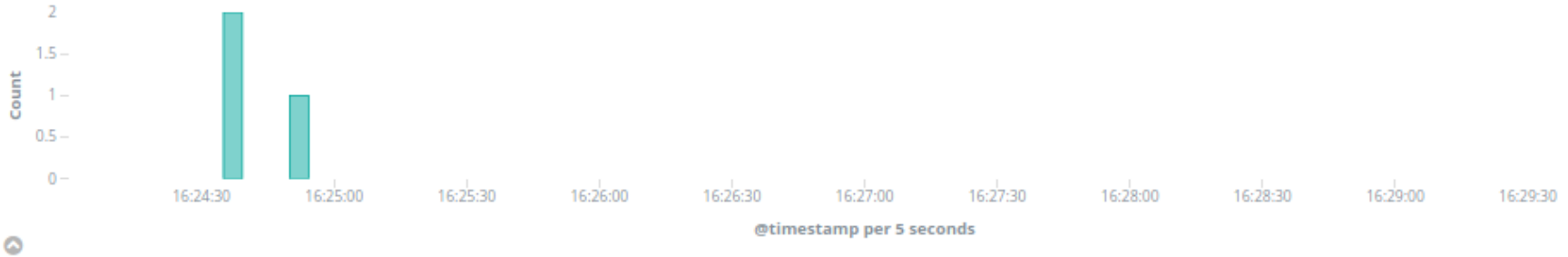
Selected Fields

? source

Available Fields

@timestamp
t AuthenticationPackageName
? Correlation
t Correlation.#attributes.Activi...
t ElevatedToken
EventCountTotal
EventIdx
EventRecordID
t GroupMembership
t ImpersonationLevel
t IpPort
KeyLength
t Keywords
Level
t LmPackageName
t LogonGuid

July 28th 2020, 16:24:00.000 - July 28th 2020, 16:30:00.000 — Auto



Time	_source
July 28th 2020, 16:24:39.742	TargetLogonId: 0x3f02704 host: [redacted] event_id: 4,624 LmPackageName: - Level: 0 @timestamp: July 28th 2020, 16:24:39.742 Correlation.#attributes.ActivityID: FD3806C7-64AC-0002-1507-38FDAC64D601 port: 49,102 client_ip: [redacted] 50 TargetOutboundUserName: - ProcessName: - SubjectLogonId: 0x0 VirtualAccount: %%1843 KeyLength: 0 TargetOutboundDomainName: - user [redacted] SubjectDomainName: - channel: security Opcode: 0 IpPort: 64166 LogonType: 3 ProcessId: 0x0 TargetLinkedLogonId: 0x0 Keywords: 0x8020000000000000 WorkstationName: - LogonGuid: A0CF5672-A28A-1FC1-FE67-F61B6775A3AF
July 28th 2020, 16:24:39.742	TargetLogonId: 0x3f02704 Security: - Keywords: 0x8020000000000000 host: [redacted] event_id: 4,627 Level: 0 @timestamp: July 28th 2020, 16:24:39.742 Correlation.#attributes.ActivityID: FD3806C7-64AC-0002-1507-38FDAC64D601 port: 49,102 Task: 12,554 GroupMembership: [redacted]
July 28th 2020, 16:24:54.914	TargetLogonId: 0x3f02704 Task: 12,545 Keywords: 0x8020000000000000 host: [redacted] event_id: 4,634 EventRecordID: 221,670 Level: 0 @timestamp: July 28th 2020, 16:24:54.914 user: UPRI6463 TargetDomainName: [redacted] Correlation: - port: 49,102 channel: security Opcode: 0 TargetUserSi [redacted] 5 LogonType: 3 Security: - _id: _z5slHYB7ZlcxLwqLLdu _type: doc _index: lab-windows-security _score: -

SOME BASIC SEARCH

EXPECT ZERO MATCH OTHERWISE INVESTIGATE

- Explicit credential for logon : except me and myself (and some service account…) , nobody should logon explicitly => expect 0 event

event_id:4648 AND -user:<WorktstationName>$ AND -user:<CUID> AND -user: UMFD-0 AND -user: UMFD-1 AND -user: DWM-1 AND -user:SVC-mcm AND -usrr:UMFD-2 AND -user: DWM-2 AND -user: Svc-FT-FR-MCM-NetAcc

- Logon : except me and myself, nobody should do a network logon => expect 0 event

event_id:4624 AND LogonType: 3 AND -user: <WorktstationName>$ AND -user: :<CUID> AND -user:"ANONYMOUS LOGON

- Nobody should do failed logon except me => Expect 0 event

event_id:4625 AND user:?* AND -user: :<CUID> AND channel:security

- No RDP on my workstation => Expect 0 event

event_id:4624 AND LogonType: 10

- No event clearing log => Expect 0 event

event_id:1102 AND channel:security

- No one added to a security-enabled local group => Expect 0 event

event_id:4732 AND channel:security

- Process creation

event_id:4688 AND channel:security AND -ParentProcessName:"C:\\Windows\\System32\\smss.exe" AND -ParentProcessName:"C:\\Windows\\System32\\wininit.exe" AND -NewProcessName:Registry AND -NewProcessName:"C:\\Windows\\System32\\smss.exe"

NEXT LEVEL SEARCH

EXPECT TO INVESTIGATE

- Service creation => more than 100 results

event_id:7045 AND channel:system AND -ImagePath:"System32\\drivers“

- Getting rid of McAfee and update day => less than 20 results

event_id:7045 AND channel:system AND -ImagePath:"System32\\drivers" AND -service_name:mcafee AND -@timestamp:"2020-03-06"

AUTOMATISE THESE SEARCHES

ELASTALERT IS HERE FOR THAT…

- Network_logon_event.yaml

```
alert:
- debug

description: logon on W[redacted]$ by someone else than the legitimate user

filter:
- query:
   query_string:
    query: "event_id:4624 AND LogonType:3 AND -user:W[redacted]$ AND -user:[redacted] AND -user:\"anonymous logon\""

index: lab-windows*

name: logon_alert

priority: 4

realert:
 minutes: 0

type: any
```

```
tags: [
    "logon",
    "alert_data"
]
user:[redacted]

INFO:elastalert:Skipping writing to ES: {'hits': 1, 'matches': 1, '@timestamp': '2020-12-27T11:47:37.910213Z', 'rule_name': 'logon_alert', 'starttime': '2020-01-01T00:00:00Z', 'endtime': '2020
-12-31T00:00:00Z', 'time_taken': 3.9960718154907227}
INFO:elastalert:Ran logon_alert from 2020-01-01 00:00 UTC to 2020-12-31 00:00 UTC: 1 query hits (0 already seen), 1 matches, 0 alerts sent
```

TIPS

ALIAS

- Alias

```
#grep IP
alias grepip='grep -P "\d{1,3}\.\d{1,3}\.\d{1,3}\.\d{1,3}« ’

# JQ .EventID, sort and Uniq them
alias jqsu='jq .EventID | sort | uniq’
```

- FONCTION with params

Retrive_element by event id :

```
#!/bin/bash

Retrieve_item() {
   cat $1 | jq ‘. | select (.EventID==$2) ’
}

Retrieve_item
```

ANALYSING WINDOWS LOG

TO GO FURTHER

- Atomic
  "Atomic Red Team is a library of simple tests that every security team can execute to test their controls"
  - https://github.com/redcanaryco/atomic-red-team

- Threat hunter playbook
  - https://threathunterplaybook.com/
  - https://github.com/OTRF/ThreatHunter-Playbook

- Windows EVTX Samples [200 EVTX examples]
  This is a container for windows events samples associated to specific attack and post-exploitation techniques
  - https://github.com/sbousseaden/EVTX-ATTACK-SAMPLES/

ANNEXES

INSTALLING EVTX_DUMP ELF VERSION

OPTION 1

- Default evtx_dump elf need libc2.18 whereas libc2.17 is installed
- Bypass is to install rust :
  curl https://sh.rustup.rs -sSf | sh
- Do the build :
  cargo install evtx
- Test : evtx_dump –h

OPTION 2

- Take the binary I built for you ;) on centOS 7

Evtx_dump in rust :
https://github.com/omerbenamram/evtx/releases