Building a Cloud-Based SIEM with Elastic Stack

Author: ZtotheZ
Degree Program: Bachelor of Engineering, Information Communications Technology
Course: Operational Security
Study Number: <redacted>

Project Overview

To develop a **cloud-based SIEM solution**, I set up **Elastic Stack (ELK) on a cloud-hosted platform** to monitor and analyse security events. This project followed the core principles of **log collection, threat detection, and real-time alerting**, ensuring that security data from multiple sources was effectively processed.

The **Elastic Stack trial version** was used for this setup, hosted in the cloud via **Azure**, though the same approach can be replicated on **CentOS 7** or a resolute on-premises machine. The project aimed to explore **home-based SIEM deployment**, modifying existing methodologies to tailor the system for real-world security monitoring scenarios.

To ensure flexibility in deployment, **various Beats agents—**including **Packetbeat, Winlogbeat, Sysmon, Filebeat, and Auditbeat—**were installed on Windows 10 and CentOS 7 virtual machines. This setup allowed me to **track system activity, capture logs, and enrich collected data** using **GeoIP information** for advanced threat analysis.

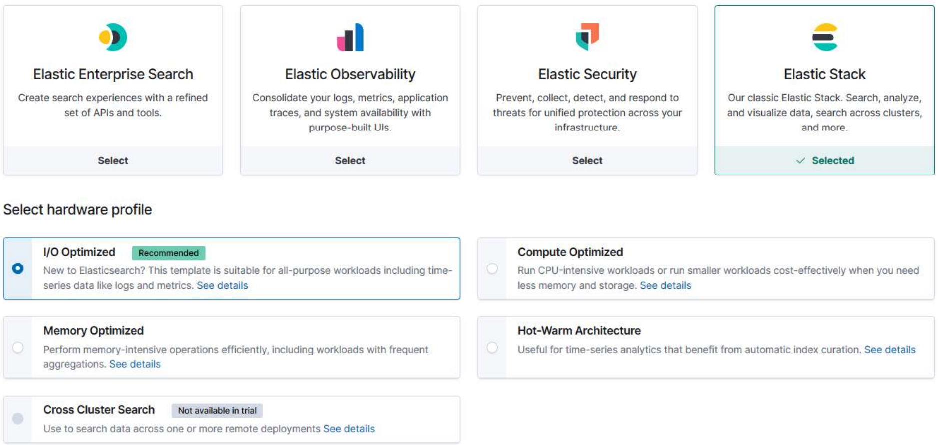
Once Elastic Stack was deployed, the next step was to configure **Kibana**, the primary interface for visualization and security monitoring. Instead of using pre-loaded sample datasets, I opted to explore the system independently, configuring **ingest pipelines, security roles, and real-time monitoring dashboards**.

Deploying Elastic Stack in the Cloud

The Elastic Stack deployment was initiated by selecting the **I/O Optimized configuration**, which provided an efficient balance between performance and scalability. While **hot-warm architecture** is a recommended setup for advanced users, the I/O Optimized approach offered a **straightforward and practical solution** for this project.

After deployment, I carefully **saved the Elastic Stack user credentials**, ensuring they were securely stored in **cloud storage solutions like OneDrive or Google Drive** to avoid data loss in case of hardware failures. Since this SIEM was cloud-based, I selected the **North Europe (Ireland) server** for optimal performance, though in a production environment, choosing a data centre closer to end users would provide better latency and efficiency.

Figure 1: Elastic Cloud Deployment with Kibana Access



Elastic Enterprise Search
Create search experiences with a refined set of APIs and tools.
Select

Elastic Observability
Consolidate your logs, metrics, application traces, and system availability with purpose-built UIs.
Select

Elastic Security
Prevent, collect, detect, and respond to threats for unified protection across your infrastructure.
Select

Elastic Stack
Our classic Elastic Stack. Search, analyze, and visualize data, search across clusters, and more.
✓ Selected

Select hardware profile

● I/O Optimized Recommended
New to Elasticsearch? This template is suitable for all-purpose workloads including time-series data like logs and metrics. See details

○ Compute Optimized
Run CPU-intensive workloads or run smaller workloads cost-effectively when you need less memory and storage. See details

○ Memory Optimized
Perform memory-intensive operations efficiently, including workloads with frequent aggregations. See details

○ Hot-Warm Architecture
Useful for time-series analytics that benefit from automatic index curation. See details

○ Cross Cluster Search Not available in trial
Use to search data across one or more remote deployments See details

---

Configuring Kibana and Enriching Security Data

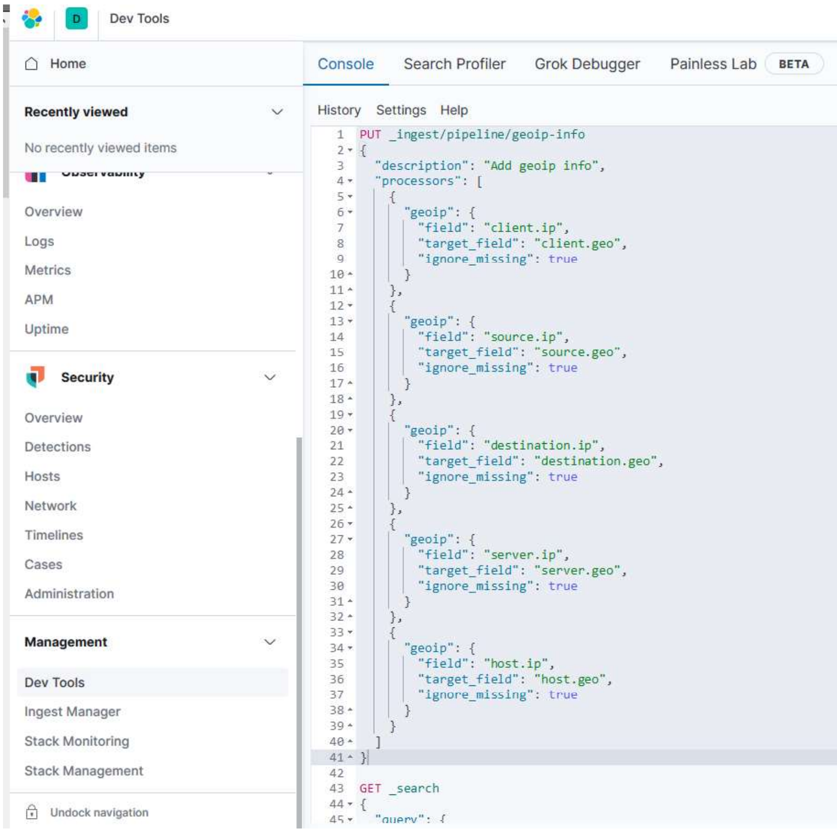
With **Kibana launched**, the next phase involved configuring **ingest pipelines** to enhance the security logs with **GeoIP data**. This enabled **geolocation tracking of security events**, which is especially useful for **detecting unauthorized access attempts from foreign locations**.

To accomplish this, I created a **custom ingest pipeline** using the **GeoIP processor**, allowing all incoming logs to be enriched with geographical metadata. This data was then indexed and made available in Kibana dashboards for analysis.

To implement this, I used the following **ingest pipeline configuration in Kibana's Dev Tools**:

```
PUT _ingest/pipeline/GeoIP-info

{

 "description": "Add GeoIP info",

 "processors": [

  { "GeoIP": { "field": "client.ip", "target_field": "client.geo", "ignore_missing": true } },

  { "GeoIP": { "field": "source.ip", "target_field": "source.geo", "ignore_missing": true } },

  { "GeoIP": { "field": "destination.ip", "target_field": "destination.geo", "ignore_missing": true } }

 ]

}
```

This pipeline automatically **appended geographical data** to logs originating from different IP addresses, making it easier to find **suspicious activity based on geographic anomalies**.



```
PUT _ingest/pipeline/geoip-info
{
  "description": "Add geoip info",
  "processors": [
    {
      "geoip": {
        "field": "client.ip",
        "target_field": "client.geo",
        "ignore_missing": true
      }
    },
    {
      "geoip": {
        "field": "source.ip",
        "target_field": "source.geo",
        "ignore_missing": true
      }
    },
    {
      "geoip": {
        "field": "destination.ip",
        "target_field": "destination.geo",
        "ignore_missing": true
      }
    },
    {
      "geoip": {
        "field": "server.ip",
        "target_field": "server.geo",
        "ignore_missing": true
      }
    },
    {
      "geoip": {
        "field": "host.ip",
        "target_field": "host.geo",
        "ignore_missing": true
      }
    }
  ]
}

GET _search
{
  "query": {
```

Figure 2: GeoIP Enrichment in Kibana

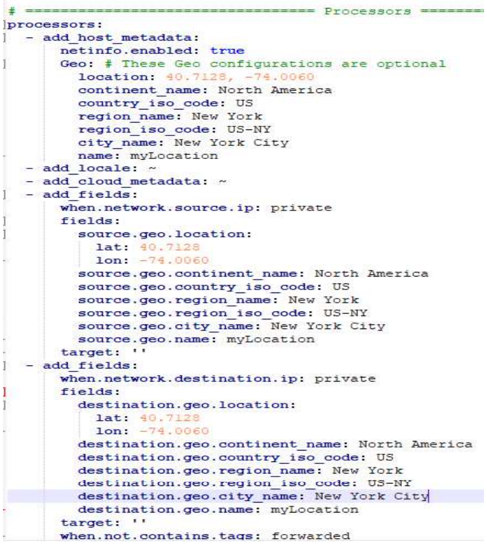
Installing and Configuring Beats Agents

To collect **host and network activity data**, I deployed **multiple Beats agents** on both Windows and CentOS machines.

- **Winlogbeat** was installed on **Windows 10 (VMware virtual appliance)** to track **Windows Event Logs**, focusing on **failed login attempts, account modifications, and PowerShell execution**.
- **Packetbeat** was set up to **capture network traffic**, enabling visibility into **potentially malicious connections**.
- **Auditbeat** was installed on CentOS 7 to **watch system events, file integrity changes, and user activity**.

During installation, I stored **all necessary authentication details (cloud ID, username, password)** in a **secure text file**, ensuring a smooth configuration process. The **Winlogbeat configuration file (winlogbeat.yml)** was changed to include **GeoIP enrichment** and **network metadata** for enhanced log analysis.

Figure 3: Winlogbeat and Auditbeat Configuration for GeoIP Processing



```
# ================================= Processors ========
processors:
  - add_host_metadata:
      netinfo.enabled: true
      Geo: # These Geo configurations are optional
        location: 40.7128, -74.0060
        continent_name: North America
        country_iso_code: US
        region_name: New York
        region_iso_code: US-NY
        city_name: New York City
        name: myLocation
  - add_locale: ~
  - add_cloud_metadata: ~
  - add_fields:
      when.network.source.ip: private
      fields:
        source.geo.location:
          lat: 40.7128
          lon: -74.0060
        source.geo.continent_name: North America
        source.geo.country_iso_code: US
        source.geo.region_name: New York
        source.geo.region_iso_code: US-NY
        source.geo.city_name: New York City
        source.geo.name: myLocation
      target: ''
  - add_fields:
      when.network.destination.ip: private
      fields:
        destination.geo.location:
          lat: 40.7128
          lon: -74.0060
        destination.geo.continent_name: North America
        destination.geo.country_iso_code: US
        destination.geo.region_name: New York
        destination.geo.region_iso_code: US-NY
        destination.geo.city_name: New York City
        destination.geo.name: myLocation
      target: ''
      when.not.contains.tags: forwarded
```

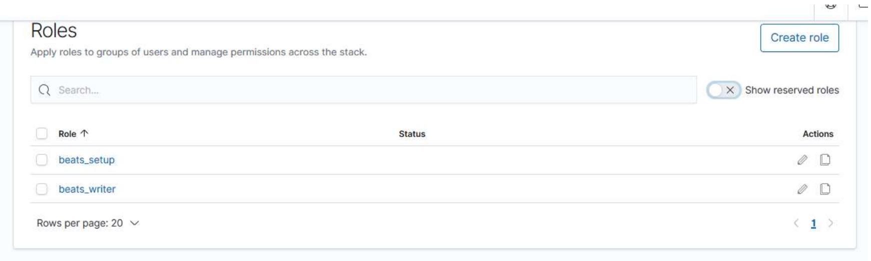
Role-Based Access Control & Security Permissions

To enforce **access control** within the SIEM system, I created **custom security roles and users** in Kibana.

- **Beats_Setup Role:** Granted permissions for managing ingestion pipelines and monitoring data sources.
- **Beats_Write Role:** Restricted to log writing without administrative privileges.
- **SIEM_User Role:** Read-only access to security data and dashboards.

By implementing **role-based access control (RBAC)**, I ensured that **privileged operations were restricted**, preventing unauthorized users from changing security configurations.

Figure 4: Role-Based Access Control in Kibana Security Settings



Roles

Apply roles to groups of users and manage permissions across the stack.

Role ↑	Status	Actions
beats_setup		
beats_writer		

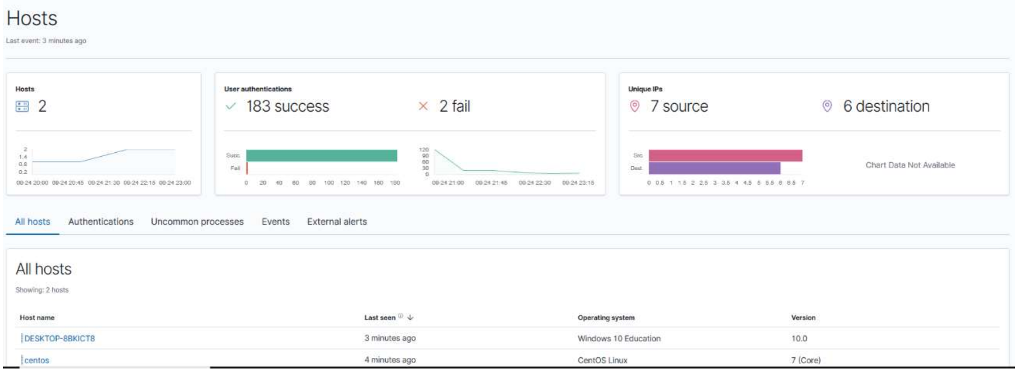
Rows per page: 20

Evaluating the SIEM with Attack Simulations

To confirm the system's effectiveness, I conducted **several attack simulations** to **generate security events and observe the SIEM's response**.

- **Brute-force SSH attack (Hydra):** A simulated brute-force attack resulted in **152 failed login attempts**, which were successfully logged, alerted, and analysed in Kibana.
- **PowerShell script execution:** Unauthorized PowerShell commands triggered security alerts, demonstrating the effectiveness of Wingbeat's event tracking.
- **GeoIP-based anomaly detection:** By simulating a login attempt from a foreign IP address, the system flagged it as a **high-risk event**, highlighting the effectiveness of geographic log enrichment.

Figure 5: SIEM Detection of Brute-Force SSH Attack in Kibana



Hosts

Last event: 3 minutes ago

Hosts: 2

User authentications: 183 success, 2 fail

Unique IPs: 7 source, 6 destination

All hosts

Showing: 2 hosts

Host name	Last seen ↓	Operating system	Version
DESKTOP-8BKICT8	3 minutes ago	Windows 10 Education	10.0
centos	4 minutes ago	CentOS Linux	7 (Core)

Final Thoughts & Future Enhancements

Deploying **Elastic Stack as a cloud-based SIEM** provided deep insights into **security events, threat detection, and automation**. The implementation demonstrated:

✓ Successful log collection from Windows & Linux hosts using multiple Beats agents.

✓ Integration of GeoIP data to enhance threat detection and location tracking.

✓ Role-based access control to enforce least privilege principles.

✓ Effective attack detection through real-world simulations (brute-force, PowerShell abuse, network monitoring).

Next Steps for Future Improvements

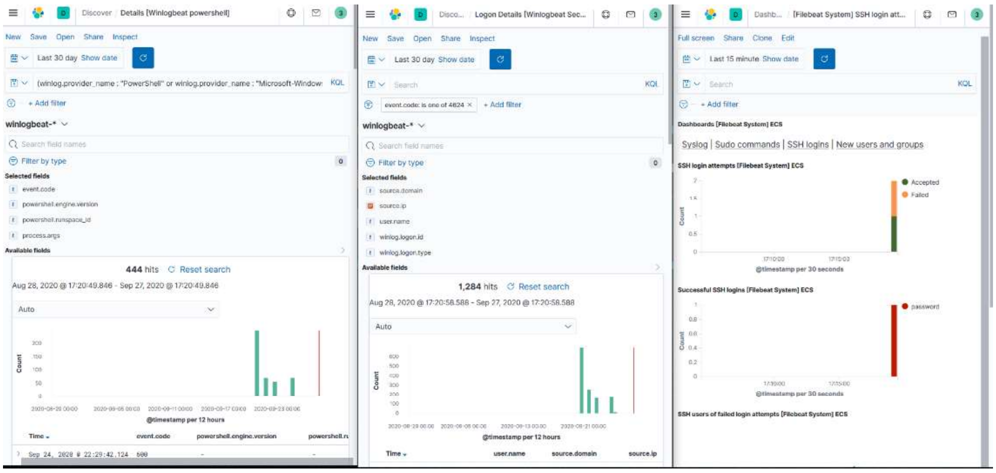
- Expanding SIEM monitoring to include **machine learning-based anomaly detection**.
- Integrating **threat intelligence feeds** to improve correlation capabilities.
- Enhancing automation for **blocking attackers via SIEM-driven security policies**.

Final Reflection

This project was an exciting opportunity to **design and implement a fully functional SIEM system**, leveraging **Elastic Stack for security monitoring**. The experience of **configuring, testing, and analysing real security events** provided valuable insights into **log analysis, automation, and cybersecurity defence strategies**.

Given more time, I would explore **deploying an Active Directory-integrated SIEM**, adding **Sysmon for in-depth process tracking**, and further enhancing **response automation using machine learning**.

Figure 6: SIEM Dashboard Showing Live Security Events



Details [Winlogbeat powershell]

Last 30 day

(winlog.provider_name : "PowerShell" or winlog.provider_name : "Microsoft-Window

winlogbeat-*

Selected fields: event.code, powershell.engine.version, powershell.runspace_id, process.args

Available fields

444 hits

Aug 28, 2020 @ 17:20:49.846 - Sep 27, 2020 @ 17:20:49.846

Time	event.code	powershell.engine.version	powershell.ru
Sep 24, 2020 @ 22:29:42.124	600	-	-

Logon Details [Winlogbeat Sec...

Last 30 day

event.code: is one of 4624

winlogbeat-*

Selected fields: source.domain, source.ip, user.name, winlog.logon.id, winlog.logon.type

Available fields

1,284 hits

Aug 28, 2020 @ 17:20:58.588 - Sep 27, 2020 @ 17:20:58.588

Time	user.name	source.domain	source.ip

[Filebeat System] SSH login att...

Last 15 minute

Dashboards [Filebeat System] ECS

Syslog | Sudo commands | SSH logins | New users and groups

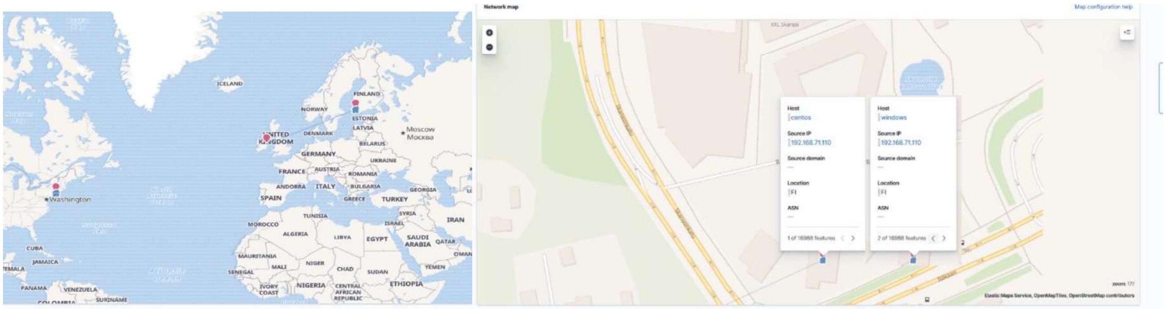
SSH login attempts [Filebeat System] ECS

Successful SSH logins [Filebeat System] ECS

SSH users of failed login attempts [Filebeat System] ECS

With the right refinements, this **SIEM setup could be scaled into an enterprise-level security monitoring solution**, providing **real-time threat detection and response capabilities for any organization**.

Figure 7: SIEM GeoIP location records



Host: centos
Source IP: 192.168.71.110
Source domain: —
Location: FI
ASN: —
1 of 16988 features

Host: windows
Source IP: 192.168.71.110
Source domain: —
Location: FI
ASN: —
2 of 16988 features

[Building-a-Cloud-Based-SIEM-with-Elastic-Stack-Config-Files]