TECH2400

Cyber Security

Workshop 10

Incident Response and Management (Part 2)

KAPLAN BUSINESS SCHOOL AUSTRALIA

Subject Learning Outcomes

LO1:	Explain the terminology associated with cyber security.
LO2:	Explain the vulnerabilities and threats pertaining to the IT infrastructure of organisations.
LO3:	Analyse risk mitigation strategies that address cyber security vulnerabilities and threats.
LO4:	Describe privacy, legal, ethical and security issues and solutions related to the IT infrastructure and use of technologies in organisations.

Weekly Schedule

Week	Topic
Week 1	Introduction and Cyber Security Foundations
Week 2	Cyber Threat Landscape
Week 3	Risk Management in Cyber Security
Week 4	Cryptography Basics and Network Fundamentals Review
Week 5	Network Security Fundamentals
Week 6	Study Success Week
Week 7	Access Control and Authentication
Week 8	Ethics and Legal Aspects of Cyber Security
Week 9	Incident Response and Management (Part 1)
Week 10	Incident Response and Management (Part 2)
Week 11	Introduction to Secure Software Development
Week 12	In-Class Assessment

Weekly Schedule

Week	Topic
Week 1	Introduction and Cyber Security Foundations
Week 2	Cyber Threat Landscape
Week 3	Risk Management in Cyber Security
Week 4	Cryptography Basics and Network Fundamentals Review
Week 5	Network Security Fundamentals
Week 6	Study Success Week
Week 7	Access Control and Authentication
Week 8	Ethics and Legal Aspects of Cyber Security
Week 9	Incident Response and Management (Part 1)
Week 10	Incident Response and Management (Part 2)
Week 11	Introduction to Secure Software Development
Week 12	In-Class Assessment

What to expect from this workshop

Last week: Hands-on experience with ELK demonstrated Detection & Analysis phase of the Incident Response Lifecycle for identifying unusual SSH failures.

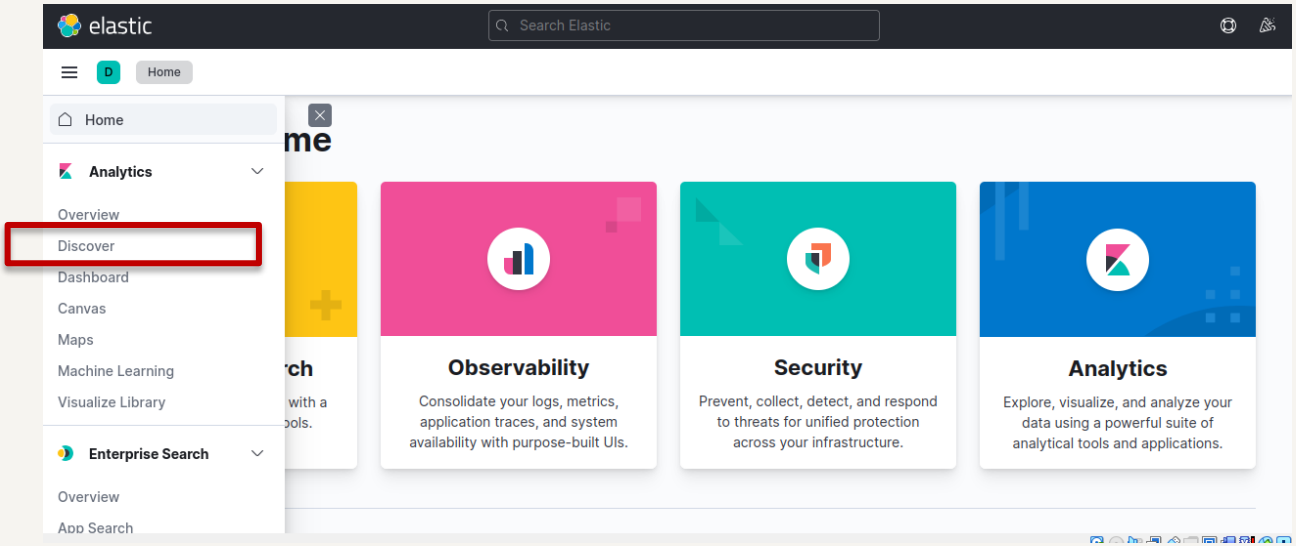
This week:

- Hands-on: More of ELK
- Network Intrusion Detection & Prevention Systems
- Hands-on: Snort on pfSense
- Post-Incident Response & Lessons Learned

ELK Activity 1

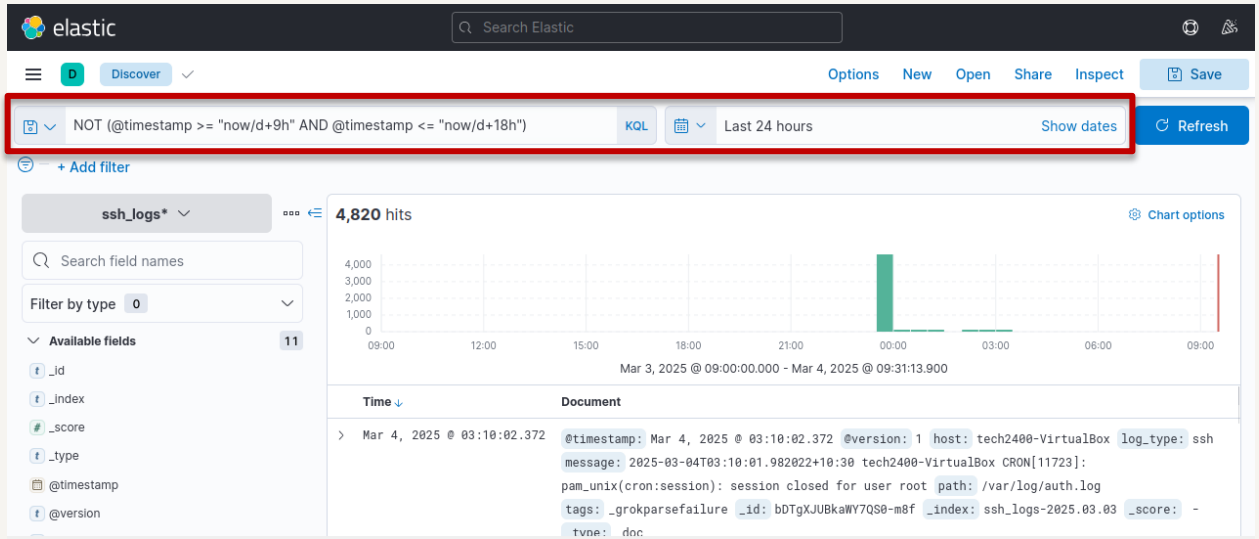
Goal: Set up an ELK log filter in Kibana to only display logs recorded after typical work hours (9AM to 6PM)

- Open Kibana (http://localhost:5601) in a browser
- Log in with your credentials
- On the left-hand sidebar, click on Discover



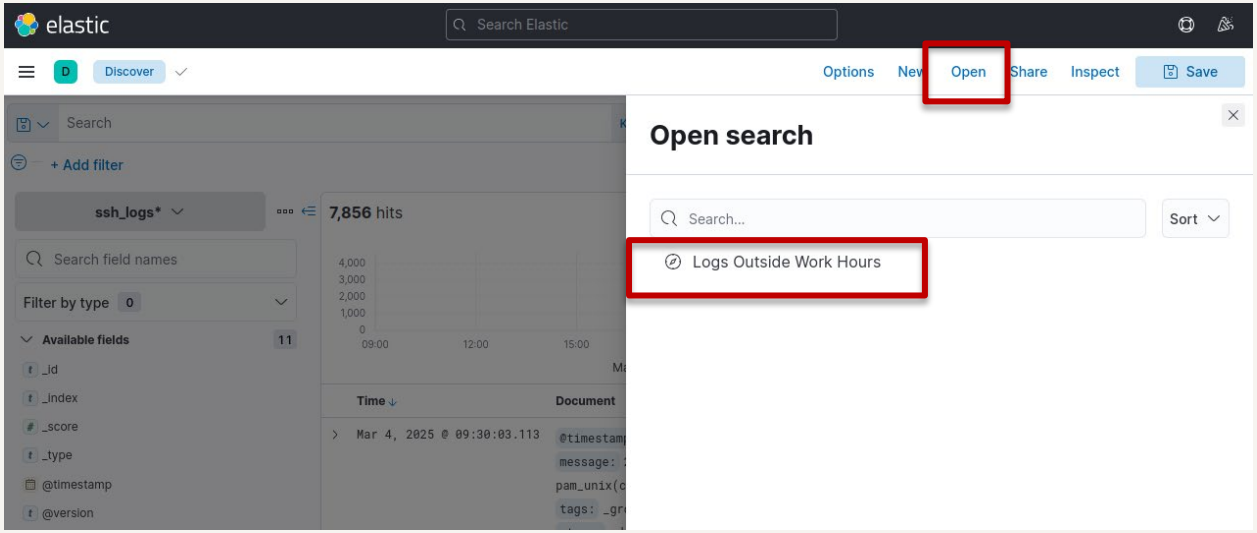
ELK Activity 1

- In the search bar at the top, enter the following query:
- NOT (@timestamp >= "now/d+9h" AND @timestamp <= "now/d+18h")



ELK Activity 1

- Click on the Save button, provide a name, and click Save
- To access, go to Discover > Open

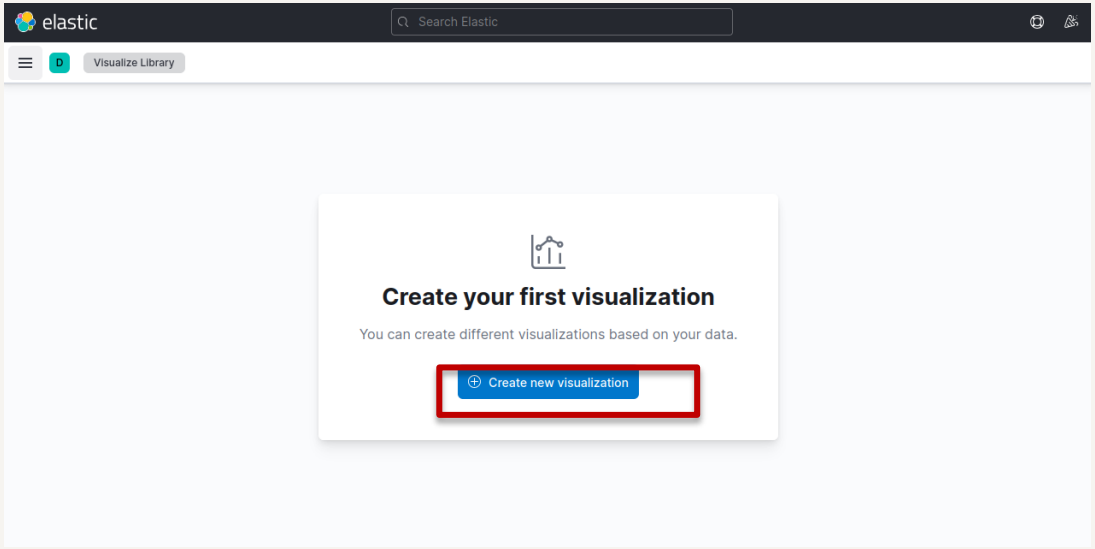


ELK Activity 2

Goal: Create visualisation to show failed login attempts over time, helping to identify patterns of attack, such as spikes at specific times.

Step 1: Select a visualisation

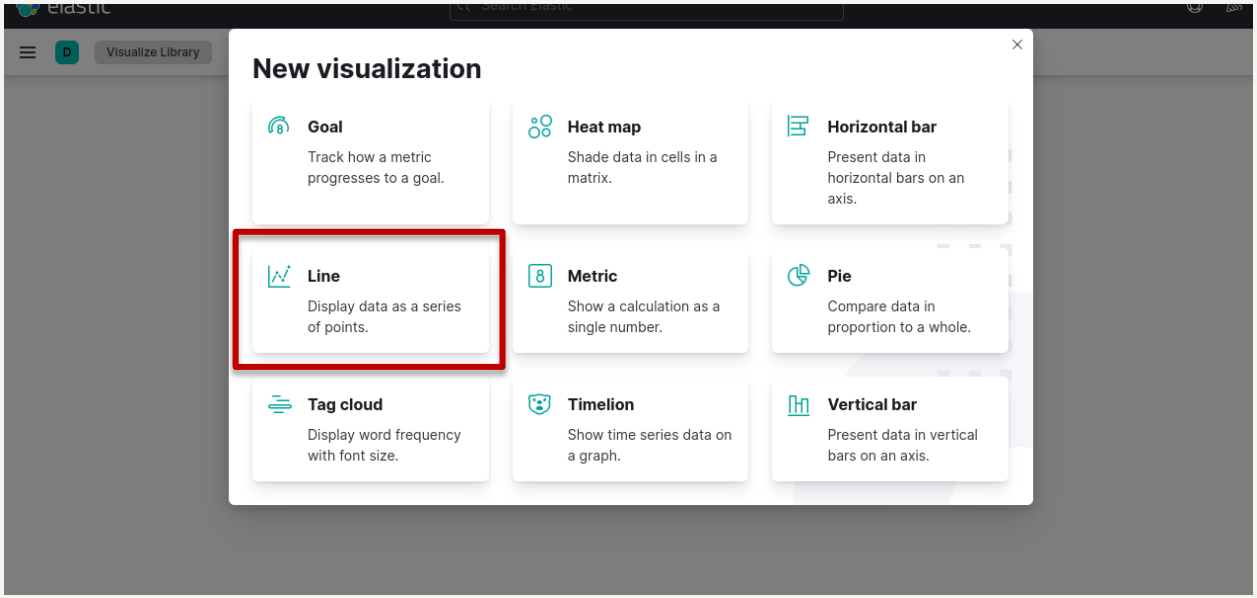
- From Kibana, navigate to Analytics > Visualize Library
- Click on Create new visualization



ELK Activity 2

Step 1: Select a visualisation

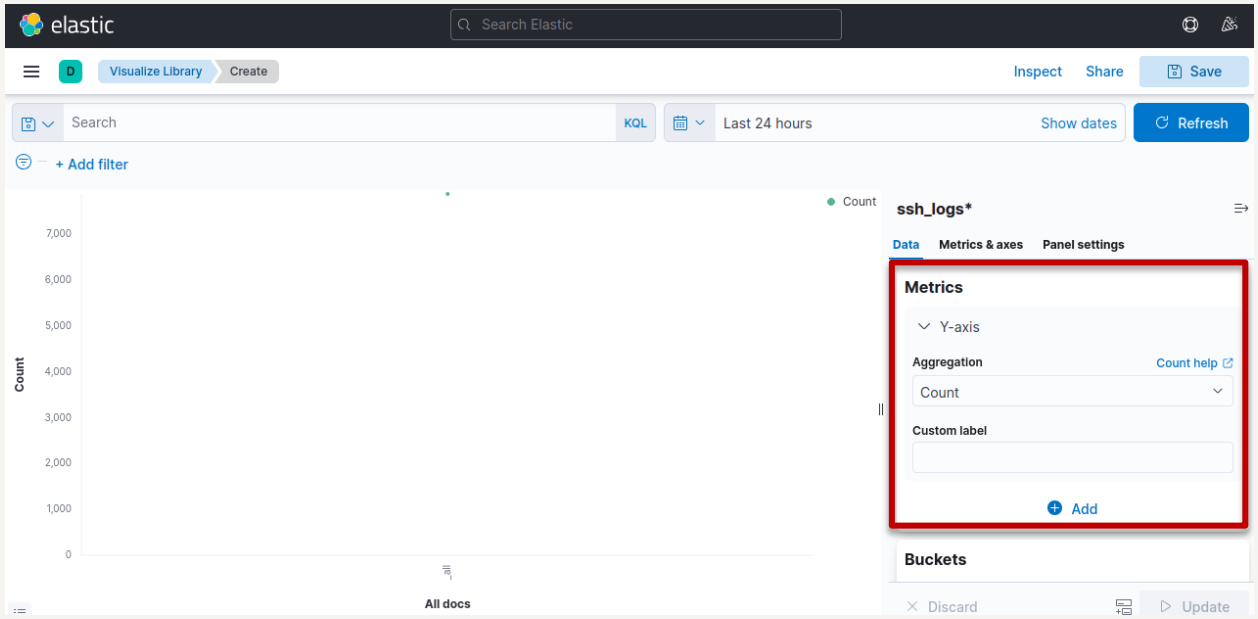
- Select Aggregation based
- Select Line
- Select ssh_logs* as an index pattern



ELK Activity 2

Step 2: Configure the visualisation

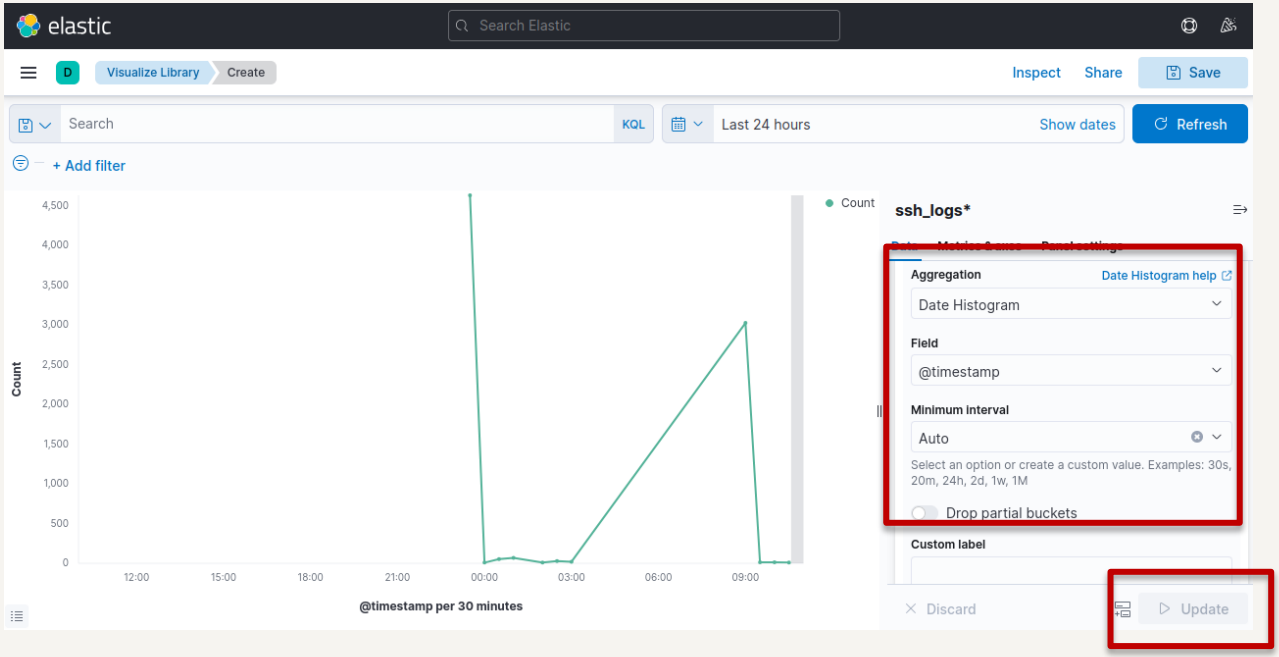
- In the Metrics section, leave the default setting:
  - Aggregation: Count
  - Field: Leave black to count all events



ELK Activity 2

Step 2: Configure the visualisation

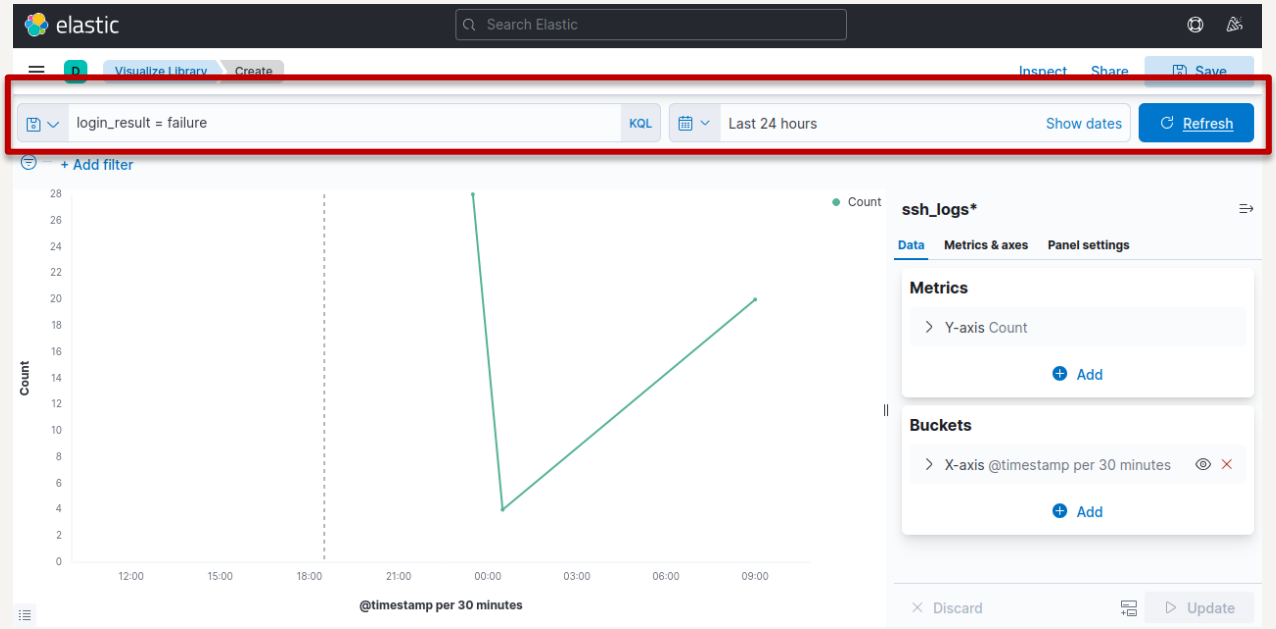
- In the Buckets section, add X-Axis
  - Aggregation: Date Histogram
  - Field: @timestamp
  - Minimum interval: Auto



ELK Activity 2

Step 3: Filter for failed logins

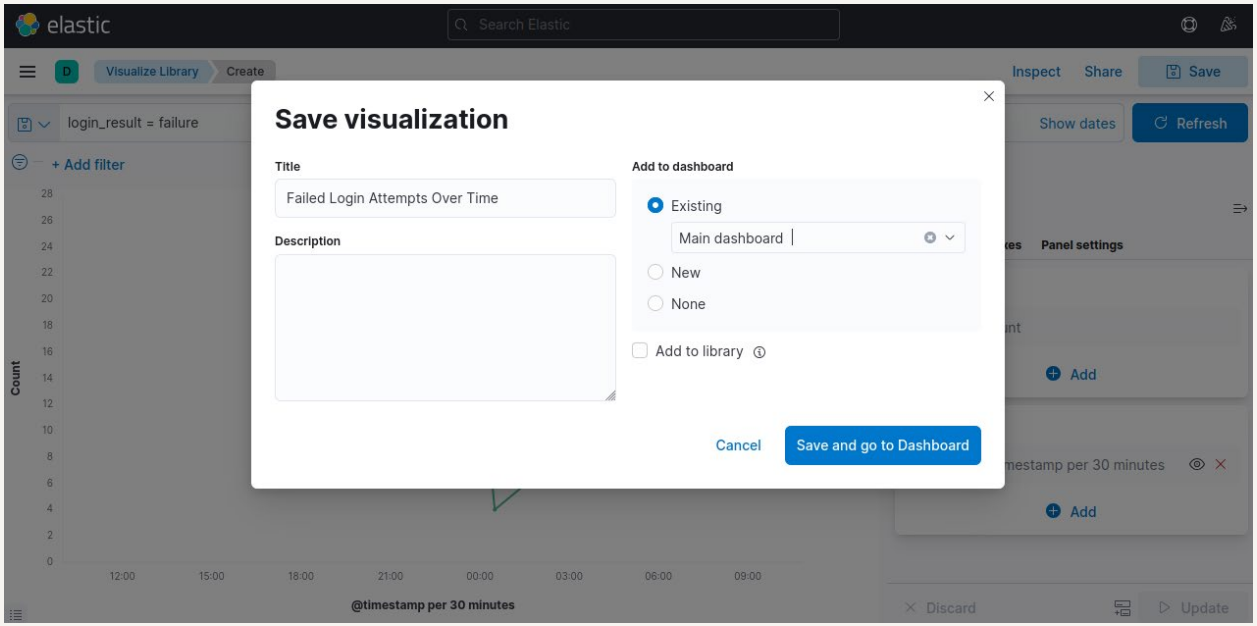
- In the search bar, enter login_result = failure
- Click Refresh



ELK Activity 2

Step 4: Save the visualisation

- Click Save
- Enter a title (e.g., Failed Login Attempts Over Time)
- Add to Main Dashboard
- Click Save and go to Dashboard



ELK Recap

ELK Stack (Log Analysis):

- Collects, stores, and visualizes log data
- Provides insights into potential security events through detailed logs and search queries
- Helps in identifying anomalies or suspicious activity during the *Detection* phase of the incident response lifecycle.

Intrusion Detection & Prevention

Intrusion Detection and Prevention (IDP) and Intrusion Detection Systems (IDS) are critical components in cyber security.

These play a key role in the *Containment* and *Eradication* phases, offering immediate responses to threats based on real-time data.

Intrusion Detection System (IDS)

An IDS is a system designed to detect and alert on unauthorised access, malicious activity, or policy violations in a network or system.

Key functions

- **Detection:** Identifies potential threats
- **Alerting:** Notifies administrators about suspicious activities
- **Reporting:** Helps generate logs for further investigation

Intrusion Prevention System (IPS)

An **IDP** combines the capabilities of IDS with active prevention, which can block or mitigate detected threats in real-time.

Key functions

- **Detection:** Like IDS, identifies suspicious activity
- **Prevention:** Takes immediate action to block or mitigate threats
- **Response:** Can automatically take steps to defend against attacks

IDS vs IPS

IDS:
- Focuses on detection and alerting
- Passive response, no action taken against detected threats

IDP:
- Detects and actively prevents threats
- Real-time response to block or mitigate attacks

Snort

Snort is an open-source network intrusion detection and prevention system (IDS/IPS)

It analyses network traffic in real-time to detect and prevent potential threats, providing robust security for networks.

Key Snort Features

- **Flexibility**: Can be used as an IDS (detection) or IPS (prevention).
- **Real-time Traffic Analysis**: Monitors network traffic and alerts or blocks suspicious activity.
- **Scalability**: Can be deployed in various environments, from small networks to large enterprise infrastructures.
- **Open Source**: Free to use with an active community contributing to continuous updates and improvements.

Key Snort Features

Rule-based Detection Methodology

- **Signature-based Detection**: Uses predefined rules (signatures) to identify known threats and attacks.
- **Anomaly-based Detection**: Detects unusual patterns or behavior in traffic that might indicate an attack, even if the specific signature is unknown.
- **Customisable Rules**: Users can write and modify rules based on their specific network environment and threat landscape.

Snort Components

1) **Preprocessors**: Prepare incoming data by normalising and categorising traffic, making it easier for the detection engine to analyse.
2) **Detection Engine**: The core of Snort, where packet analysis occurs. It applies rules to analyse traffic and detect potential threats.
3) **Output Modules**: Responsible for sending alerts or taking actions (e.g., blocking traffic) based on the detection engine's findings. Alerts can be logged or integrated with other systems for further analysis.

Snort + pfSense

Snort: Real-time IDS/IPS traffic analysis and detection

+

pfSense: Robust firewall and routing platform

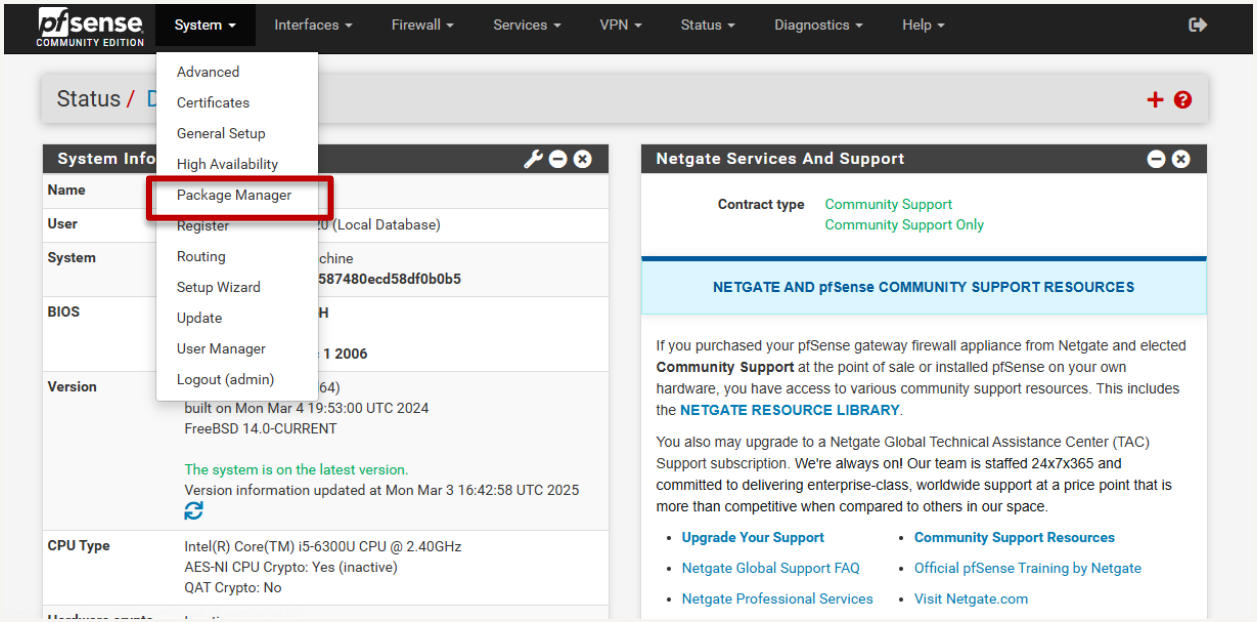
=

A powerful, integrated solution that not only detects and prevents threats but also effectively manages and filters network traffic, providing comprehensive protection.

Activity: Snort Installation

Goal: Install Snort package on pfSense

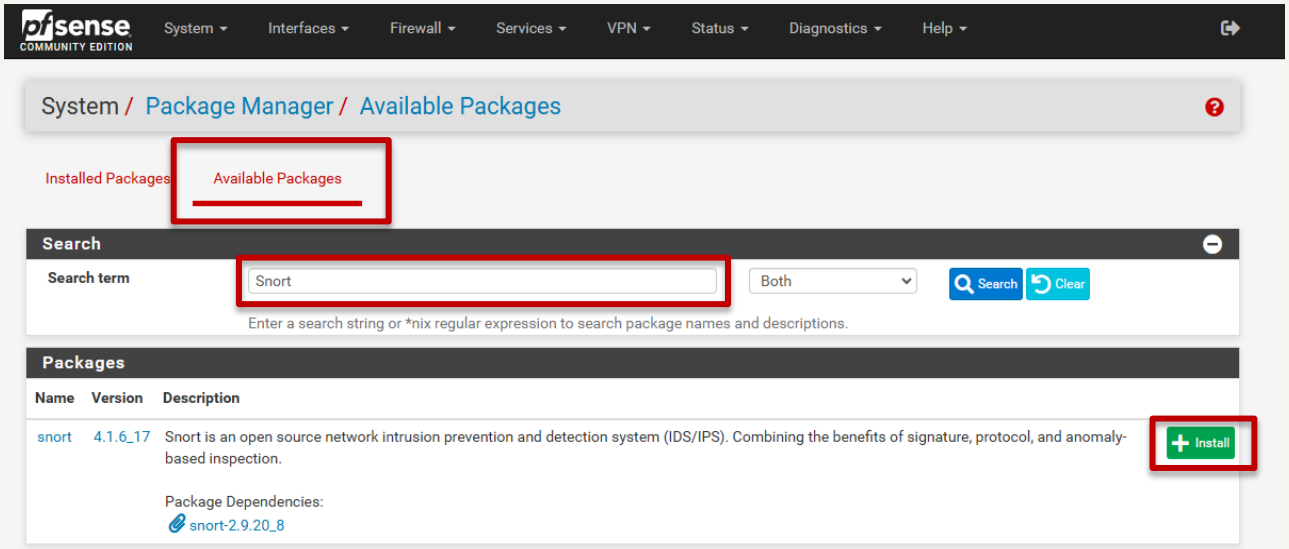
- Login to pfSense Web Interface
- Navigate to System > Package Manager



Activity: Snort Installation

Goal: Install Snort package on pfSense

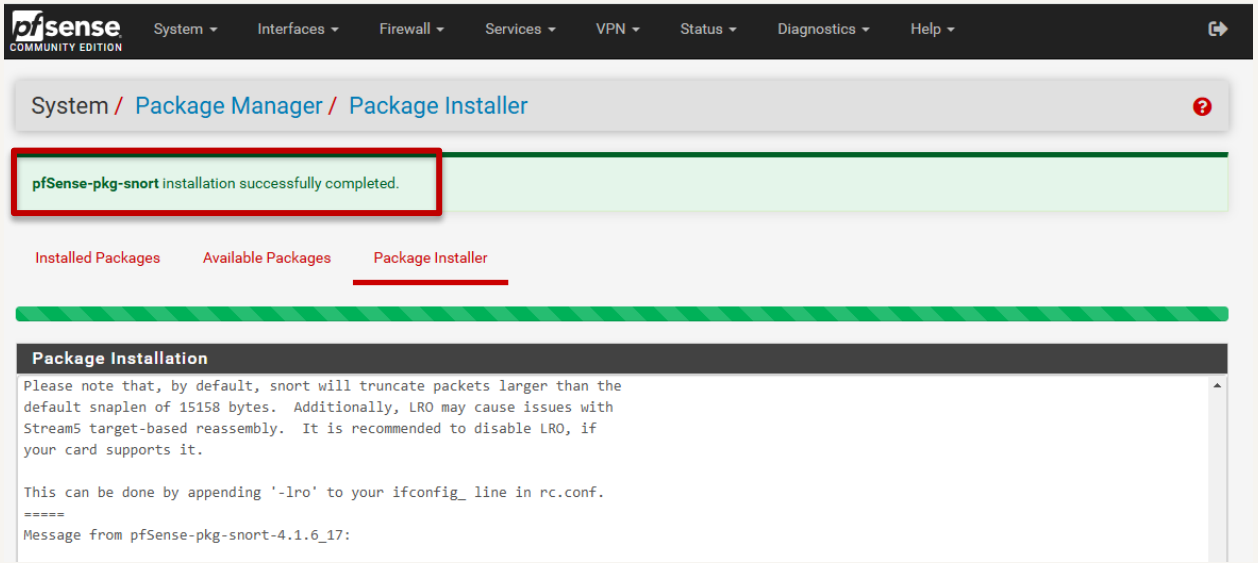
- Click on the Available Packages tab
- In the search bar, type Snort and press Enter
- Click +Install



Activity: Snort Installation

Goal: Install Snort package on pfSense

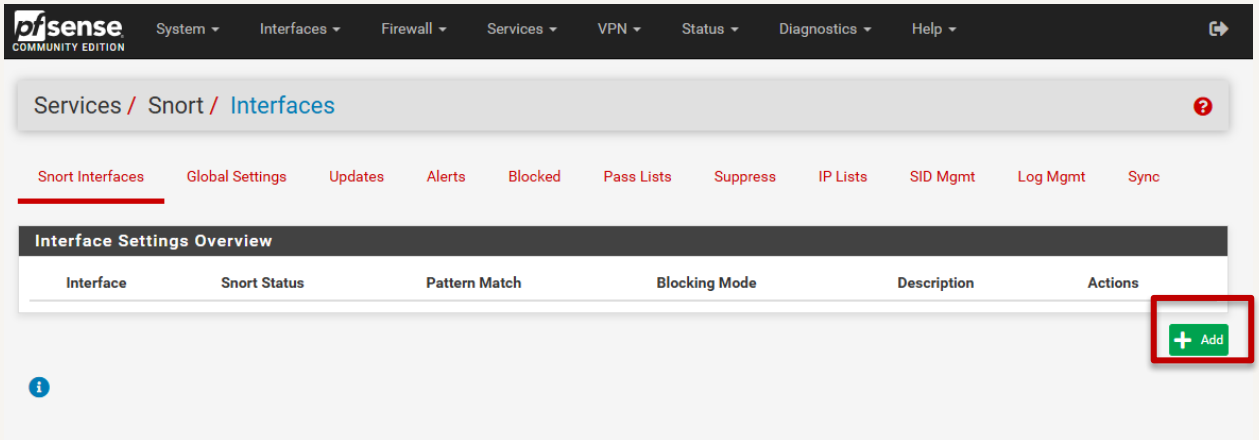
- Confirm the installation, and wait for the process to complete



Activity: Snort Installation

Goal: Enable Snort on WAN interface

- Go to Services > Snort > Interfaces
- Click on +Add

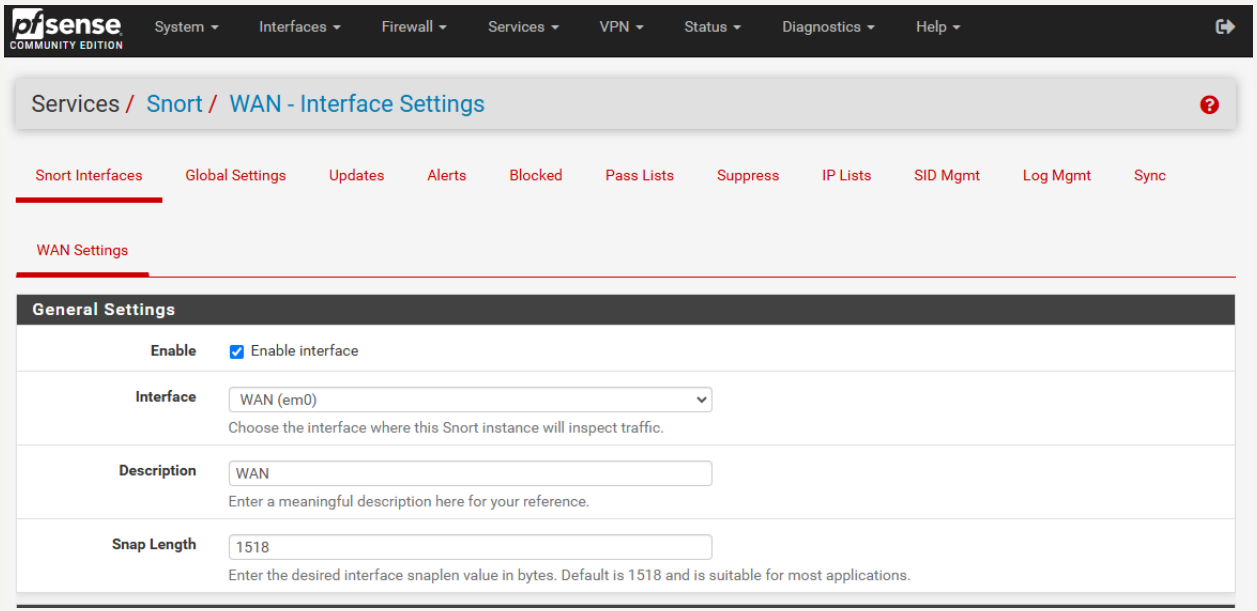


KAPLAN BUSINESS SCHOOL AUSTRALIA

Activity: Snort Installation

Goal: Enable Snort on WAN interface

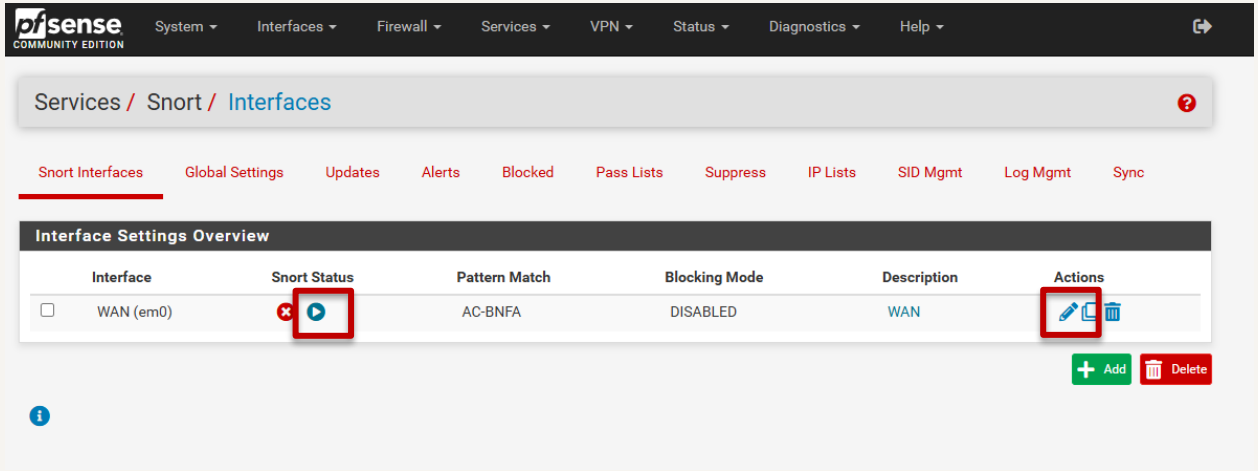
- Check the box for Enable interface
- Set the Interface to WAN (em0) in the dropdown
- Click Save at the bottom of the page



Activity: Snort Installation

Goal: Enable Snort on WAN interface

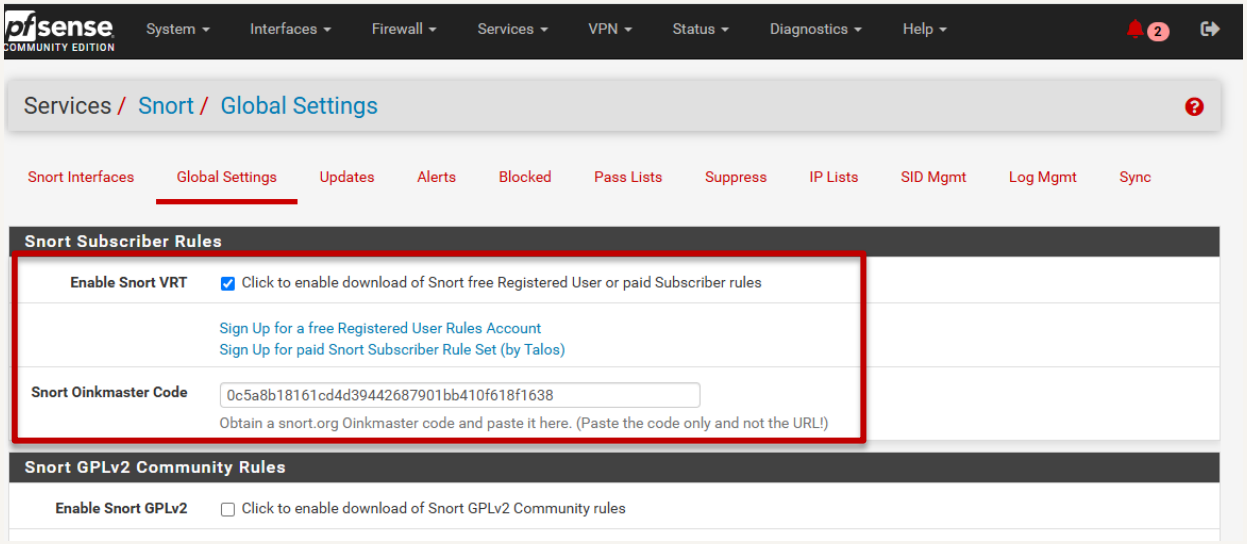
- Navigate back to Snort Interfaces
- Start Snort on WAN by clicking the blue play button
  Note: Snort Status should change to
- Click the pencil icon to edit



Activity: Snort Installation

Goal: Enable Rulesets

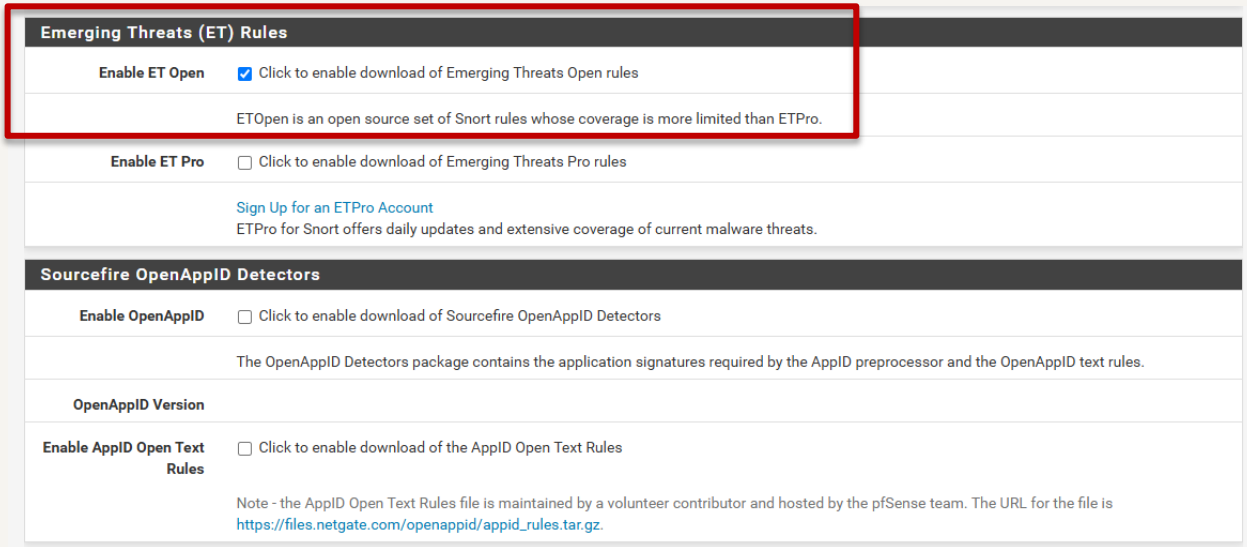
- Click on the Global Settings tab
- Enable Snort VRT
  - Note: Sign up for a free Registered User Rules account to get an Oinkmaster Code



Activity: Snort Installation

Goal: Enable Rulesets

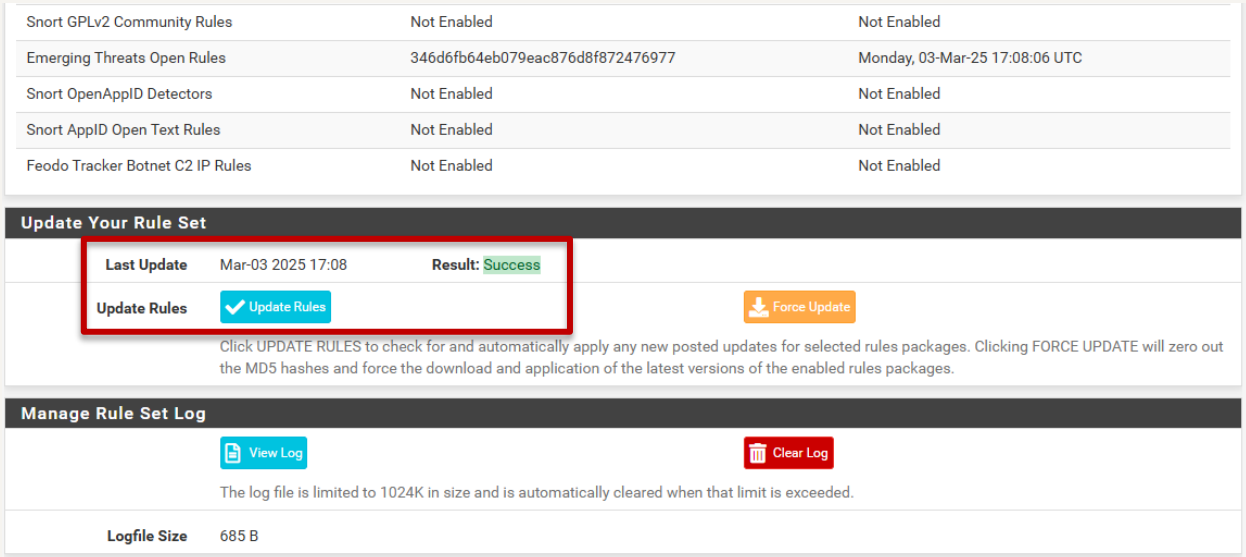
- Scroll down to Emerging Threats Rules
- Check the tick box beside "Enable ET Open"
- Click Save at the bottom of the page



Activity: Snort Installation

Goal: Enable Rulesets

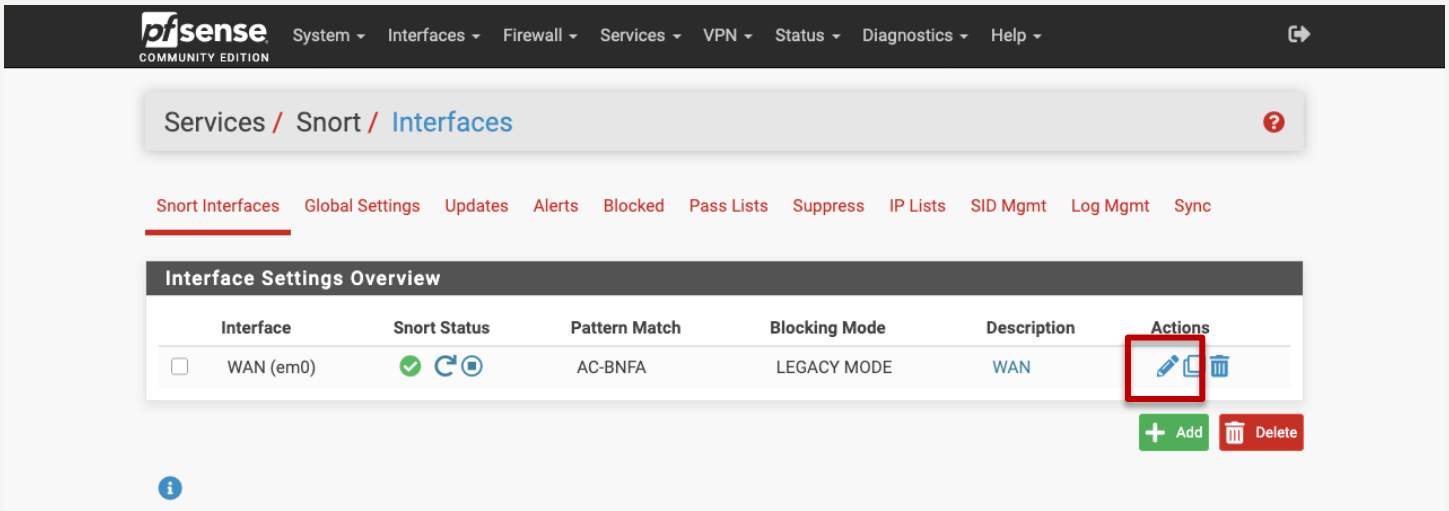
- Navigate to the Updates tab
- Click Update Rules to download the latest ruleset
- Wait for the update to complete



Snort Activity 1

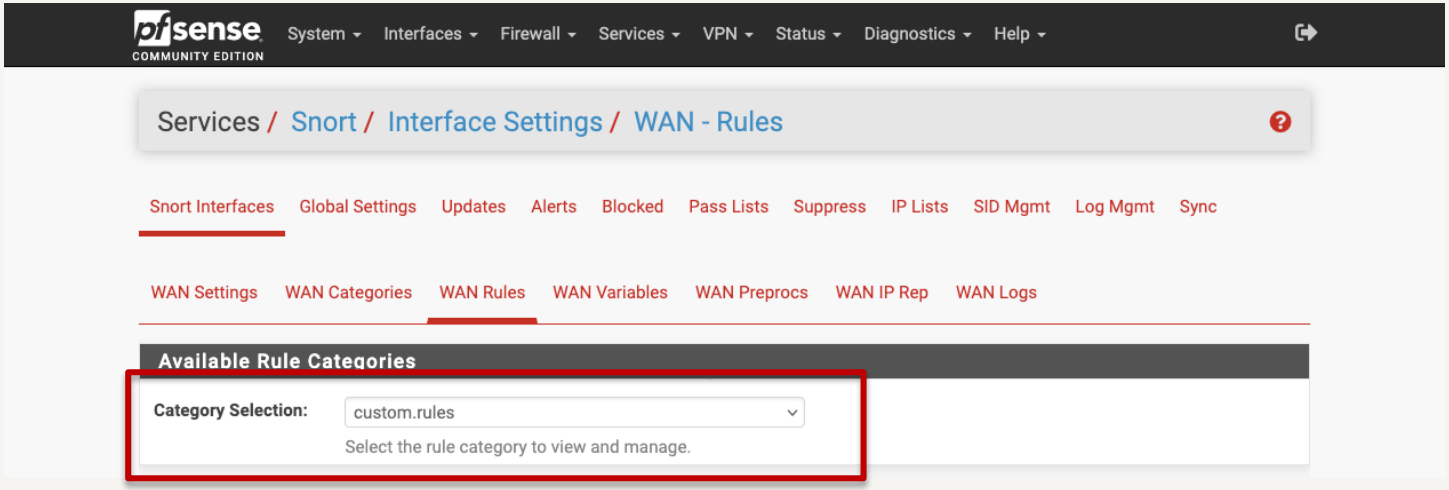
Goal: Configure Snort to trigger an alert when an HTTP GET request contains a suspicious user-agent string.

- On pfSense, navigate to Services > Snort > Snort Interfaces
- Click on the pencil icon to edit the WAN interface



Snort Activity 1

- Navigate to the WAN Rules tab
- From the Category Selection dropdown menu, select custom.rules



Snort Activity 1

- In the Defined Custom Rules text box, enter:
- alert tcp any any -> any 80 (msg:"Suspicious User-Agent detected"; content:"GET"; http_method; content:"User-Agent: EvilBot/1.0"; http_header; sid:1000001; rev:1;)

  Note: Rule breakdown on next slide

Snort Activity 1

alert	Triggers an alert when the rule matches
tcp	Applies only to TCP traffic (used for HTTP connections)
any any	The rule applies to traffic coming from any IP and any port
->	Indicates traffic flowing from source to destination
any 80	The traffic is destined for any IP on port 80
msg:"Suspicious User-Agent detected"	The message displayed in Snort logs when the rule triggers
content:"GET"; http_method	Ensures the request uses the HTTP GET method.
content:"User-Agent: EvilBot/1.0"; http_header;	Checks if the HTTP header contains User-Agent: EvilBot/1.0.
sid:1000001;	A unique identifier for this Snort rule
rev:1	The rule's version number

Snort Activity 1

- Test by opening a Command Prompt in Windows and run
- curl -A "EvilBot/1.0" http://example.com

- Verify alert in Snort logs
  - Navigate to Services > Snort > Alerts
  - Look for an alert matching "SSH connection attempt from restricted IP range"

Snort Activity 1

In real-world scenarios, suspicious User-Agent strings often belong to:

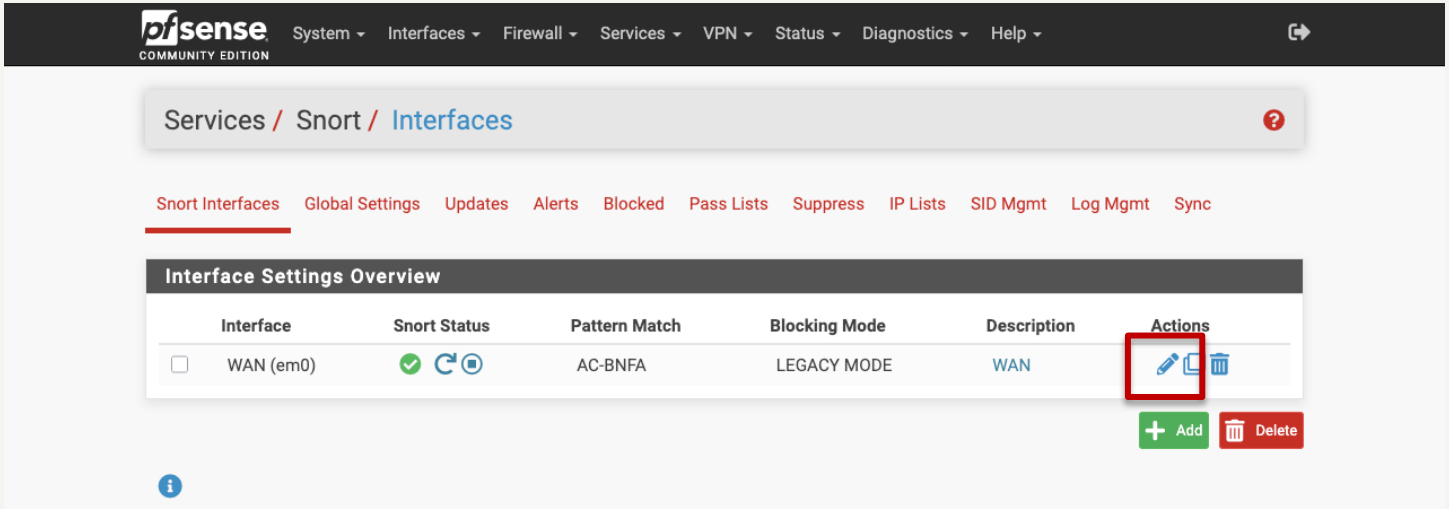
Web Scrapers & Bots:
Scrapy, Puppeteer, Googlebot, GPTbot

Common Attack Tools:
sqlmap, Nikto, Metasploit

Snort Activity 2

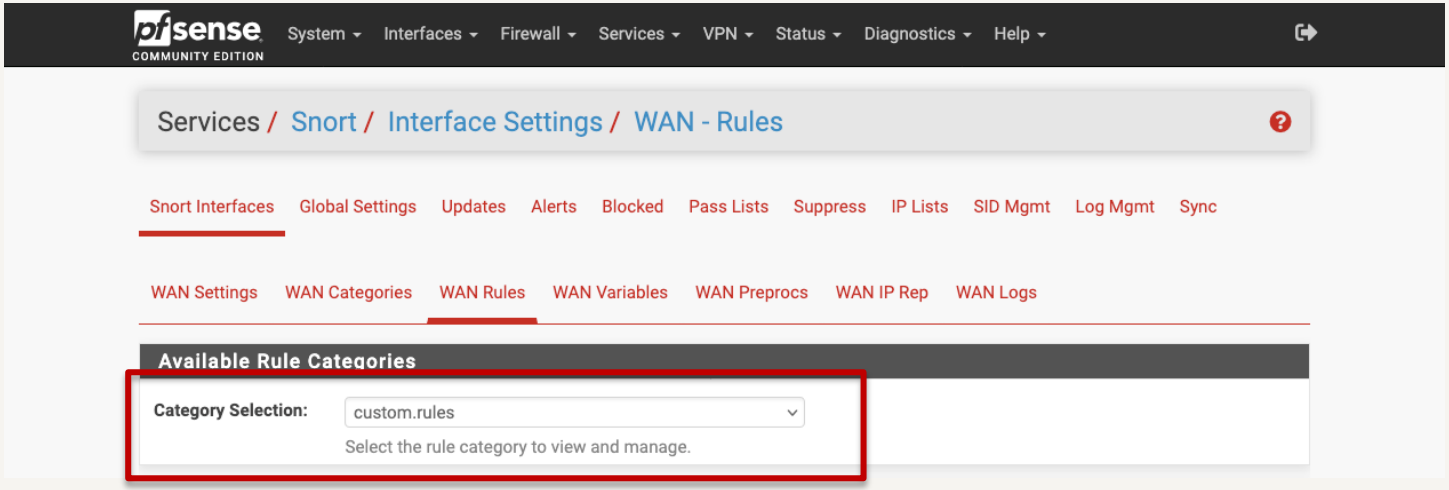
Goal: Configure Snort to trigger an alert when an SSH connection attempt is made from a specific IP address range.

- Navigate to Services > Snort > Snort Interfaces
- Click on the pencil icon to edit the WAN interface



Snort Activity 2

- Navigate to the WAN Rules tab
- From the Category Selection dropdown menu, select custom.rules



Snort Activity 2

- In the Defined Custom Rules text box, enter:
- alert tcp [192.168.1.0/24] any -> any 22 (msg:"SSH connection attempt from restricted IP range"; flow:to_server,established; sid:1000002; rev:1;)
  Note: Rule breakdown on next slide
- Click Save

Snort Activity 2

alert	Triggers an alert when the rule matches
tcp	Applies only to TCP traffic (SSH operates over TCP port 22)
192.168.1.0/24 any	Matches traffic from any IP in the 192.168.1.0/24 subnet and any source port
->	Indicates traffic flowing from source to destination
any 22	The traffic is destined for any IP on port 22
msg:"SSH connection attempt from restricted IP range"	The message displayed in Snort logs when the rule triggers
flow:to_server,established;	Ensures the rule only applies to established SSH connections directed to a server.
sid:1000002;	A unique identifier for this Snort rule
rev:1	The rule's version number

Snort Activity 2

- Test by opening a Command Prompt in Windows and run
- ssh user@<pfSense_ip>
- telnet <pfSense_ip> 22

- Verify alert in Snort logs
  - Navigate to Services > Snort > Alerts
  - Look for an alert matching "SSH connection attempt from restricted IP range"

Snort Activity 2

Real-world scenarios where this could be useful

Monitoring Internal Threats
Detect if employees or internal machines are trying to access SSH servers they shouldn’t be connecting to.

Detecting Lateral Movement in a Network Attack
If a machine inside your network is compromised, an attacker might try SSH connections to other internal systems.

Snort Activity 2

Real-world scenarios where this could be useful

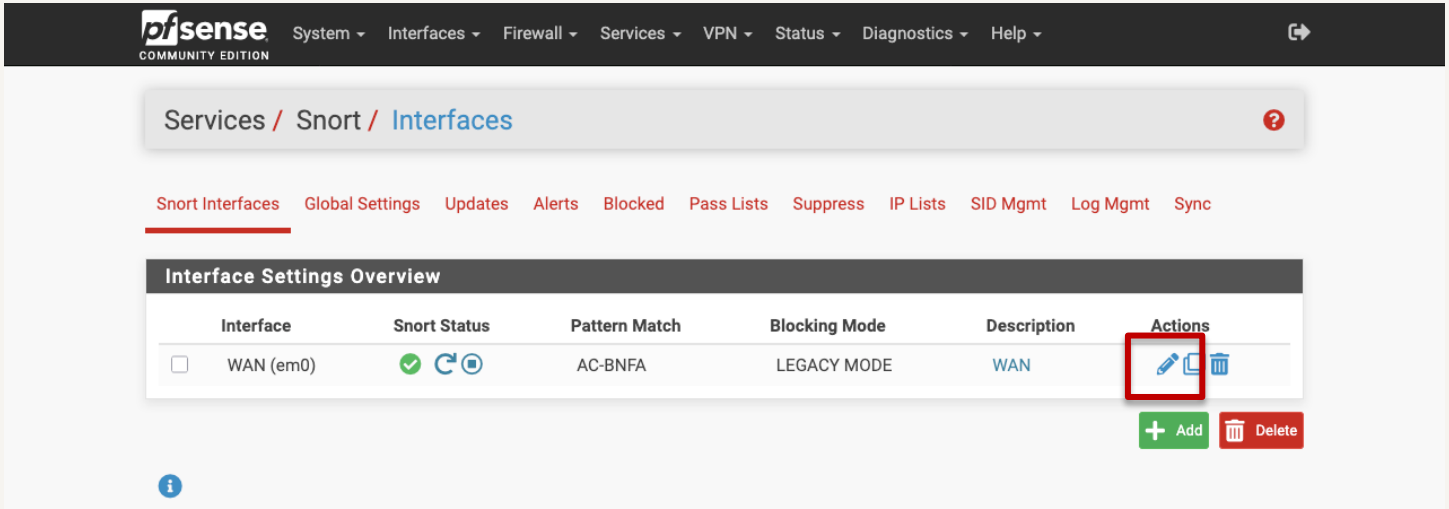
Blocking Unauthorised Remote Access Attempts
If a network segment should not have SSH access (e.g., guest Wi-Fi), alerts help detect misconfigurations or attacks.

Geofencing SSH Access
Alert when SSH access comes from certain IP ranges outside your country or organisation.

Snort Activity 3

Goal: Configure Snort to trigger an alert when it detects malicious DNS traffic.

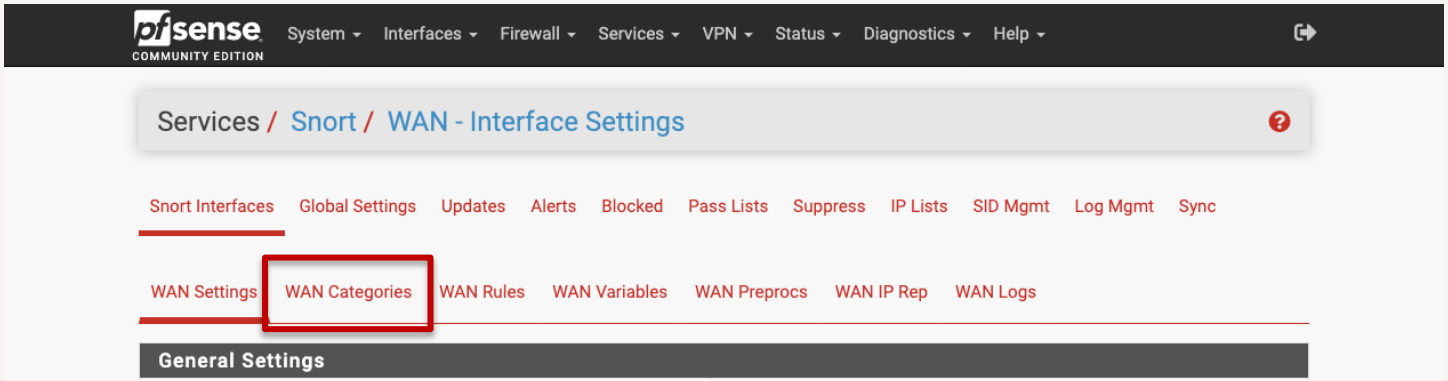
- Navigate to Services > Snort > Snort Interfaces
- Click on the pencil icon to edit the WAN interface



KAPLAN BUSINESS SCHOOL AUSTRALIA

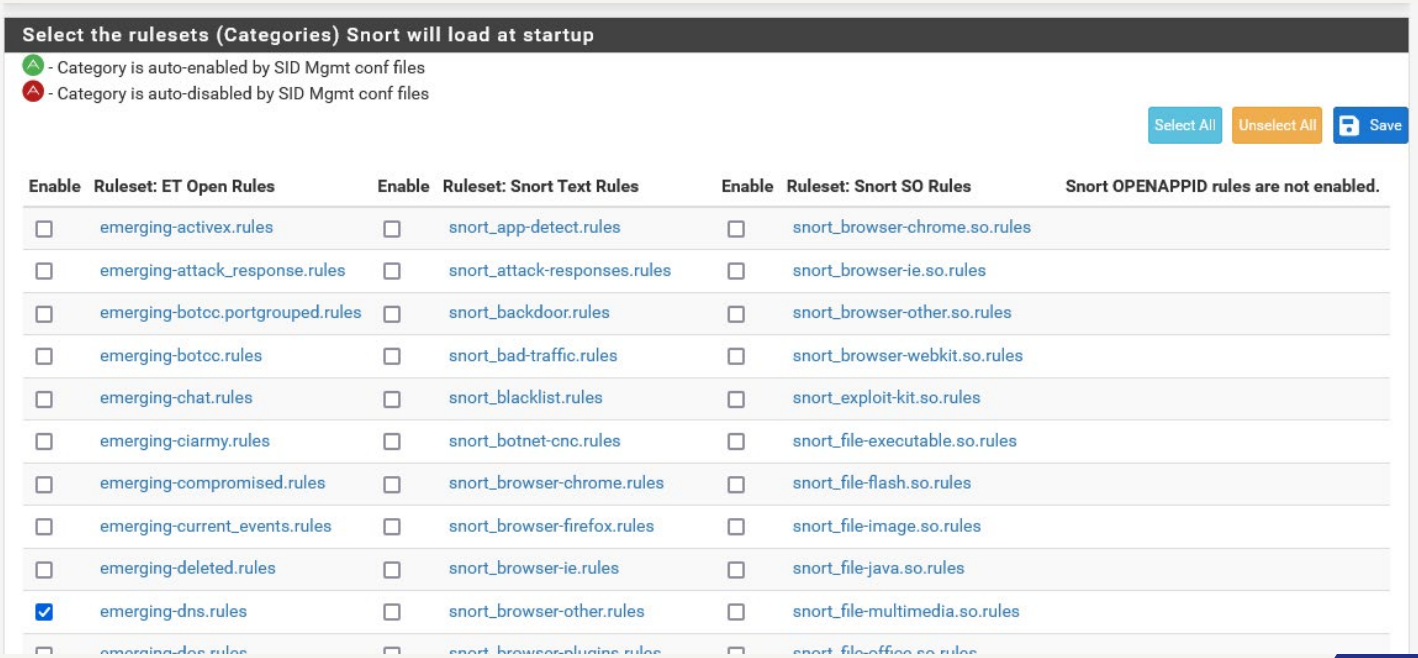
Snort Activity 3

- Navigate to the WAN Categories tab



Snort Activity 3

- Find and enable the following rules:
  - emerging-dns.rules
  - emerging-malware.rules
- Click Save



Snort Activity 3

- Test by opening a Command Prompt in Windows and run
- nslookup Vaccineprogram.co.kr

- Verify alert in Snort logs
  - Navigate to Services > Snort > Alerts
  - Look for an alert ET CNC DNS Query or similar

Snort Activity 3

Real-world scenarios where this could be useful

Monitoring Internal Threats
Detect if employees or internal machines are trying to access SSH servers they shouldn’t be connecting to.

Detecting Lateral Movement in a Network Attack
If a machine inside your network is compromised, an attacker might try SSH connections to other internal systems.

Snort Activity 3

Real-world scenarios where this could be useful

Blocking Unauthorised Remote Access Attempts
If a network segment should not have SSH access (e.g., guest Wi-Fi), alerts help detect misconfigurations or attacks.

Geofencing SSH Access
Alert when SSH access comes from certain IP ranges outside your country or organisation.

Next Week

Week 11: Introduction to Secure Software Development

- Overview of Software Development Lifecycle (SDLC)
- OWASP Top 10 Vulnerabilities
- Principles of Secure Coding
- Tools and frameworks that assist in secure software development