HITB SECCONF 2024 BANGKOK

HTTPS://CONFERENCE.HITB.ORG/HITBSECCONF2024BKK

A Design and Development of a Multi-Tenant SIEM: A Prototype Case Study using Security Onion

Wararit Hongkamnerd

Student, Kasetsart University

KU KASETSART UNIVERSITY FACULTY OF ENGINEERING

ANRES APPLIED NETWORK RESEARCH LAB.

#HITB2024BKK

HITB-TBCERT CommSec Track

29 AUG

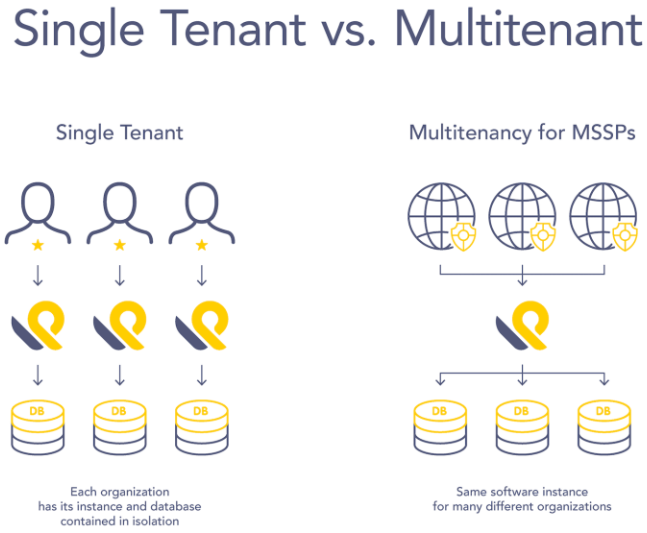
Overview

- Introduction
- Problems
- Methodology
- Results
- Key Takeaways

Introduction

Multi-tenancy

- **Single instance of software** serving multiple customers
- **Shared infrastructure** with isolated data and resources for each customer
- **Enhanced scalability** and **cost-efficiency** compared to dedicated instances
- **Data privacy** and **security** are critical considerations


Single Tenant vs. Multitenant
Single Tenant
Multitenancy for MSSPs
DB
Each organization has its instance and database contained in isolation
Same software instance for many different organizations

https://www.logpoint.com/wp-content/uploads/2020/10/single-tenant-vs.-multitenant-infographic.png

Security Onion

- **Open-source SIEM platform** for comprehensive log management and correlation
- Built on a foundation of free and open-source tools, integrated with built-in tools

Security Onion

ORY

redis

SALTSTACK

SURICATA®

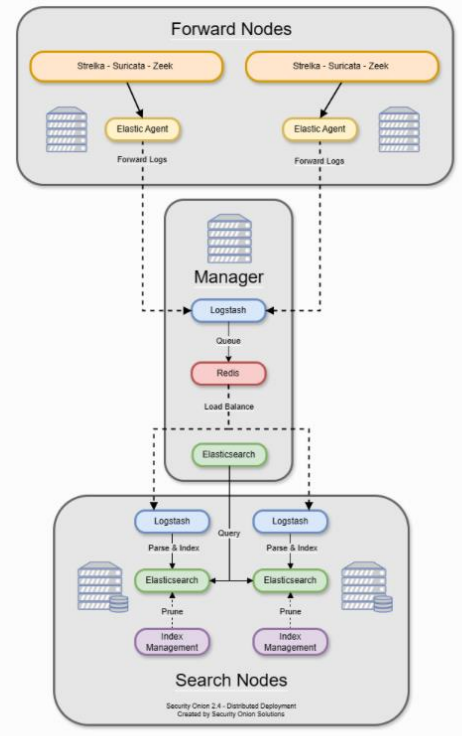
NGINX

influxdb

elastic

Security Onion

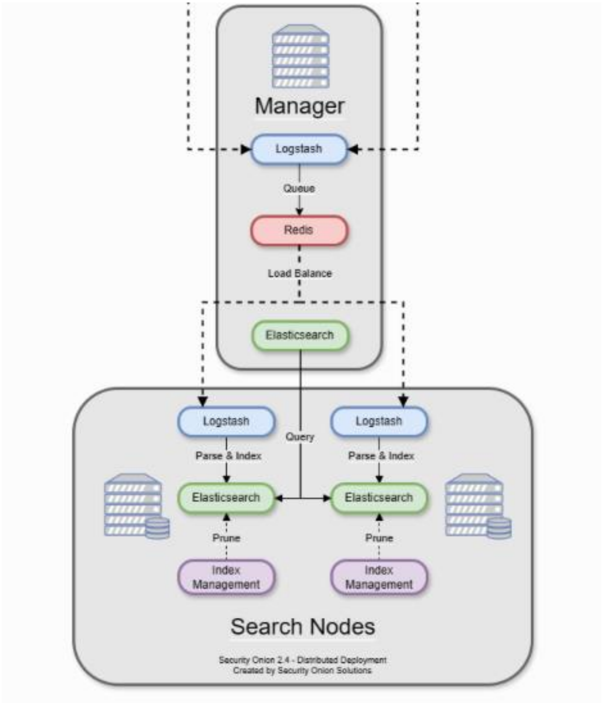
- Deployment models:
  - Standalone
  - Distributed
  - etc.



https://docs.securityonion.net/en/2.4/architecture.html

Security Onion

- Deployment models:
    - Standalone
    - Distributed
    - etc.



https://docs.securityonion.net/en/2.4/architecture.html

MSSP

- Stands for **M**anaged **S**ecurity **S**ervice **P**rovider
- External organization that provides security services for clients
- An important tool is SIEM

Problems

SIEM for MSSP

- **Multi-tenant architecture** is crucial for MSSPs to efficiently manage multiple clients' security data in isolation
- Open-source SIEMs often lack built-in multi-tenant capabilities

Time to recover SIEM

- **SIEM** is a critical tool for maintaining overall security
- Downtime of a SIEM system can significantly impact an organization's security posture
- Rapid recovery is essential to minimize risk exposure

Methodology

Solutions

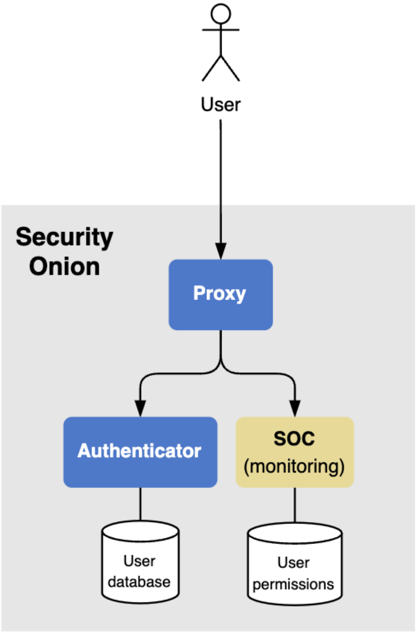
- Propose a new Security Onion architecture to support multi-tenant functionality
- Develop a separate system to manage user permissions among tenants
- Identify key factors that impact recovery time

Solution 1: A new architecture

- Propose a new architecture to support multi-tenant functionality
- Prioritize minimal modifications to the default architecture

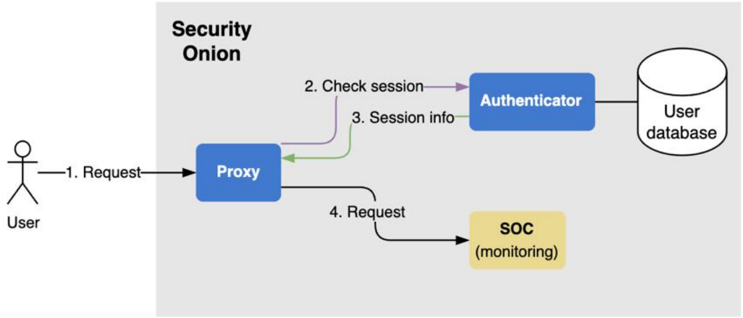
Authentication & Authorization components

- Web-based monitoring page called **SOC**
- **Kratos** as an authenticator
- Access through **Nginx** proxy



Authentication & Authorization process

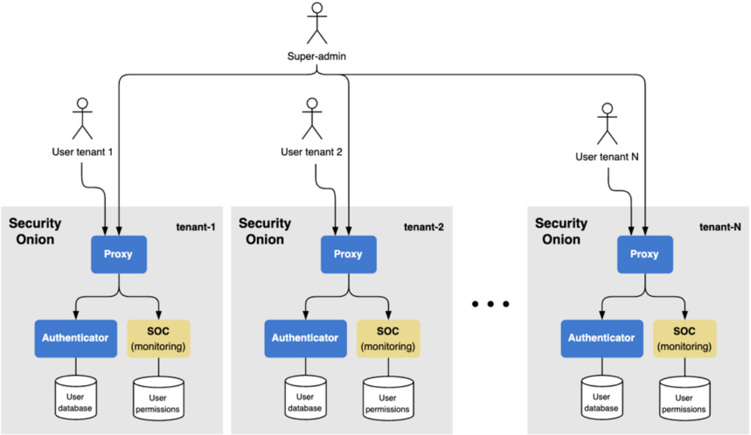
1. **User** sends a request to the **SOC** inside Security Onion
2. **Nginx** proxy creates sub-request and send to the **authenticator** to validate user's session
3. **Kratos** authenticator validates the session and returns session info
4. **Proxy** sends original request with session info for the **SOC** to authorize the user



#HITB2024BKK

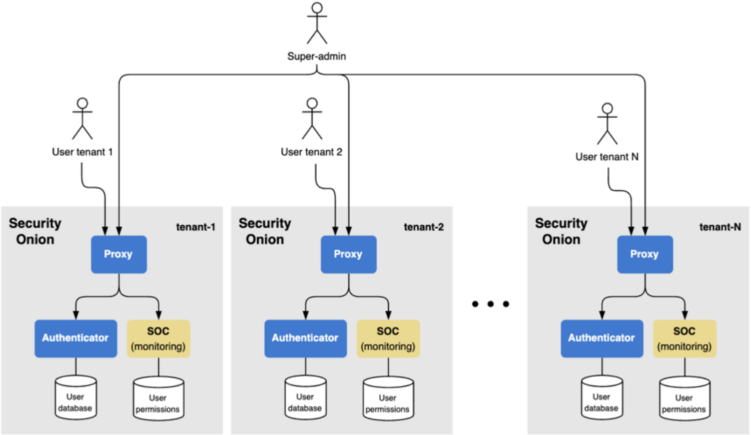
Multi-tenant architecture ?

- Each client has separated Security Onion instance
- Reduce risk of data leakage



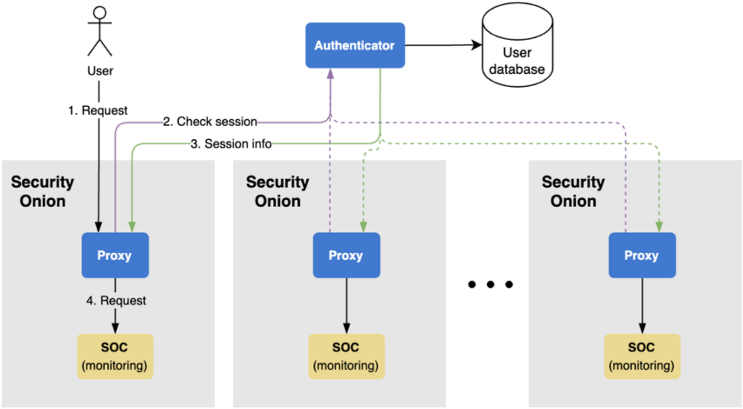
Problems

- Each tenant has its own authentication
- Distinct session token
- Users like admins need to login for each tenant



To solve the problems

- **Centralized Authentication:** Use a single login system
- **Shared Database:** Store user data in one place
- **Proxy Configuration:** Adjust proxies to use the shared authenticator
- **Shared Tokens:** Allow users to stay logged in across different tenants



#HITB2024BKK

Solutions

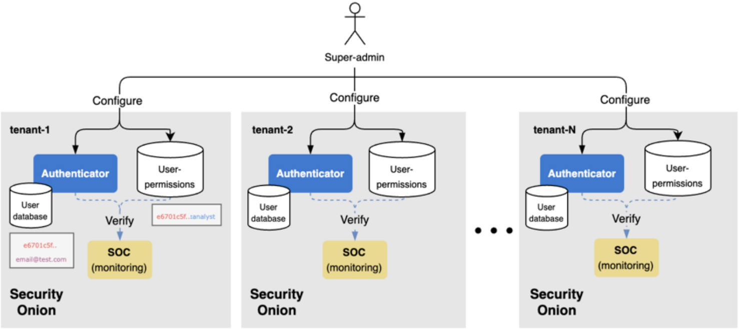
- Propose a new Security Onion architecture to support multi-tenant functionality
- Develop a separate system to manage user permissions among tenants
- Identify key factors that impact recovery time

Solution 2: Dedicated SIEM management

- While high isolation levels are crucial for security, they can increase management complexity
- A dedicated management system can simplify user authorization and access control across multiple tenants

Default permission management process

- **Authorization process:** SOC authorizes users based on
  - User ID
  - User’s role
- **User Lookup:** Queries user ID using the session token
- **Permission Modification:** Edits the user-permission file in Security Onion for the specified ID

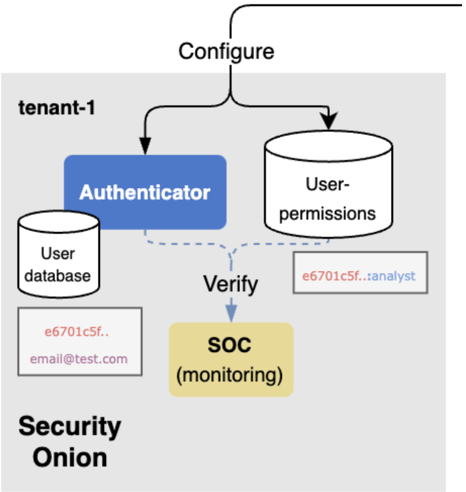


Note: **so-allow** command will manage the above steps

Default permission management process

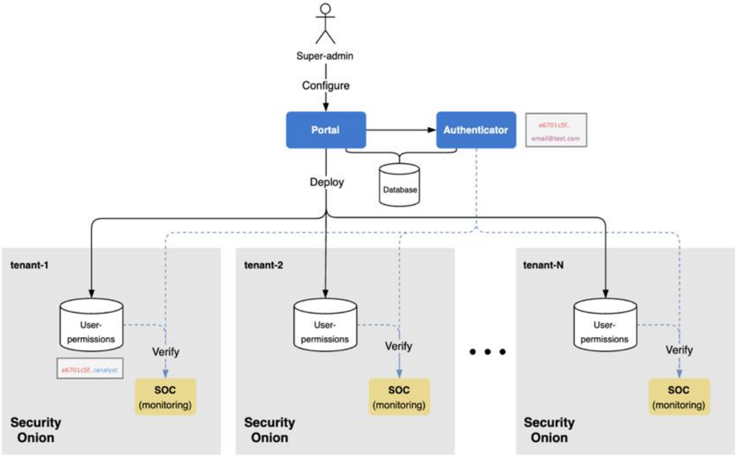
- **Authorization process:** SOC authorizes users based on
  - User ID
  - User's role
- **User Lookup:** Queries user ID using the session token
- **Permission Modification:** Edits the user-permission file in Security Onion for the specified ID

Note: **so-allow** command will manage the above steps

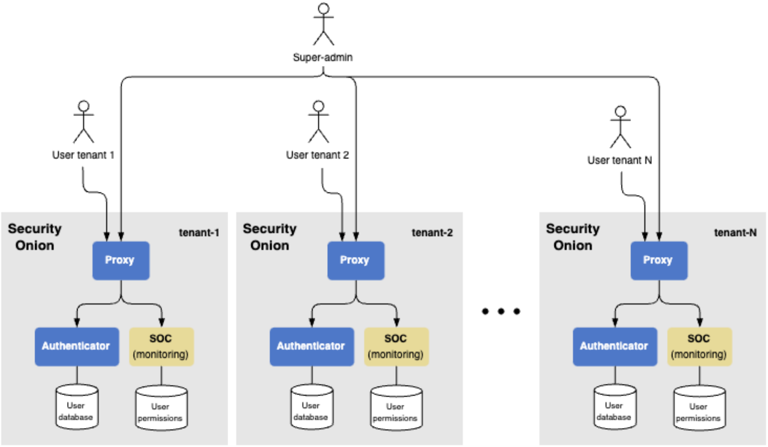


Custom permission management process

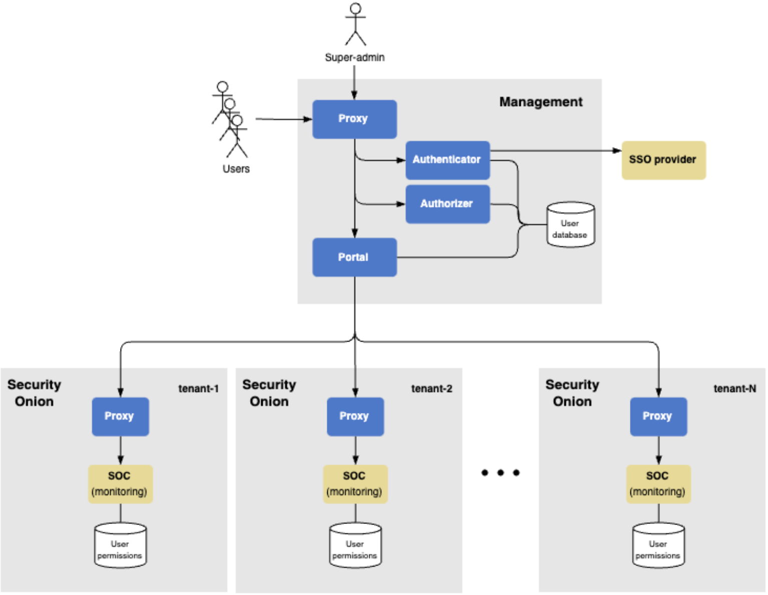
- For centralized permission management
- **Centralized Control:** Admins manage user permissions from a dedicated management system (Portal)
- **Permission Deployment:** The portal deploys permission changes to Security Onion



Comparison

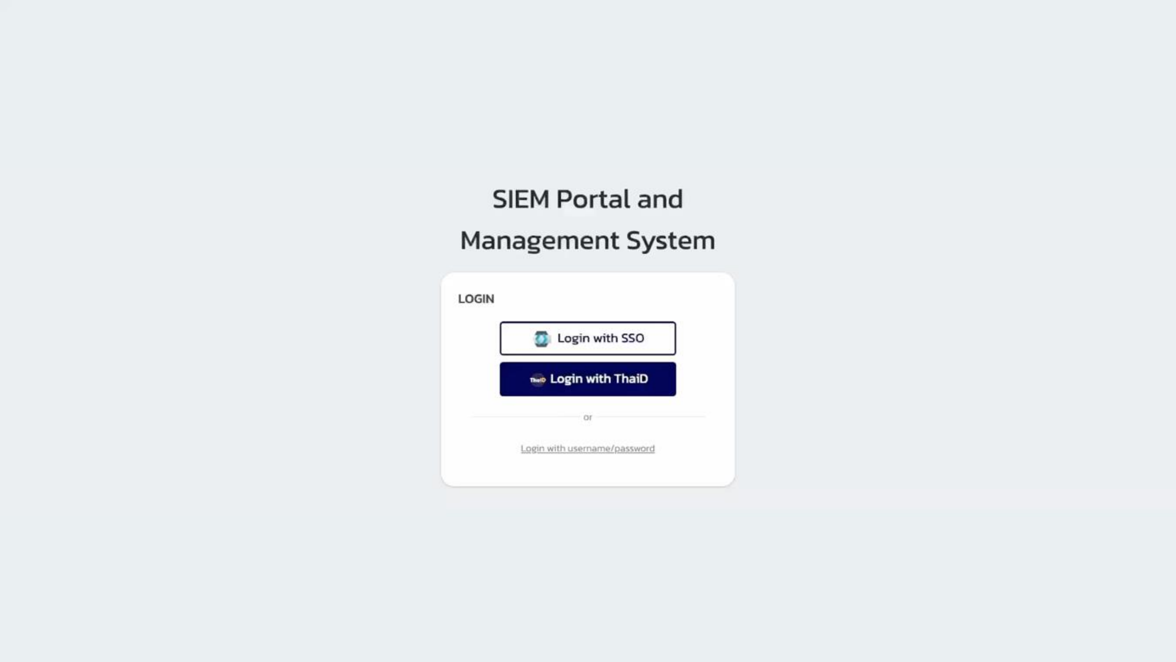


Default architecture



Multi-tenant architecture
with the management system

Demo



SIEM Portal and Management System

LOGIN

Solutions

- Propose a new Security Onion architecture to support multi-tenant functionality
- Develop a separate system to manage user permissions among tenants
- Identify key factors that impact recovery time

Solution 3: Key factors impact recovery time

- Critical for minimizing downtime and maintaining security
- Enhances operational efficiency and overall security posture

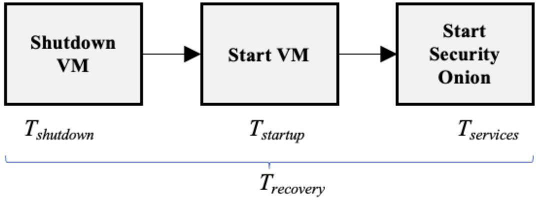
Recovery process

Recovery approach depends on issue severity

- Restart services
- **Reboot**
- **Restore from backup**

Recovery process

- Reboot



Shutdown VM

Start VM

Start Security Onion

$T_{shutdown}$

$T_{startup}$

$T_{services}$

$T_{recovery}$

$$T_{recovery} = T_{shutdown} + T_{startup} + T_{services}$$

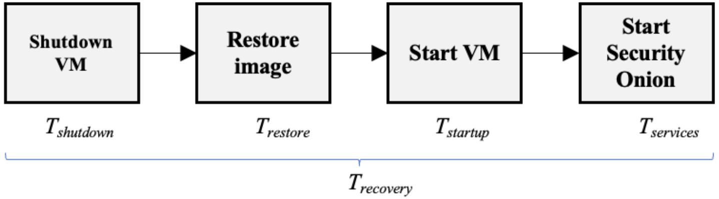
- Restore from backup

Recovery process

- Reboot

$$T_{recovery} = T_{shutdown} + T_{startup} + T_{services}$$

- Restore from backup



$$T_{recovery} = T_{shutdown} + T_{restore} + T_{startup} + T_{services}$$

Recovery process

- Reboot

$$T_{recovery} = \cancel{T_{shutdown}} + T_{startup} + T_{services}$$

- Restore from backup

$$T_{recovery} = \cancel{T_{shutdown}} + \cancel{T_{restore}} + T_{startup} + T_{services}$$

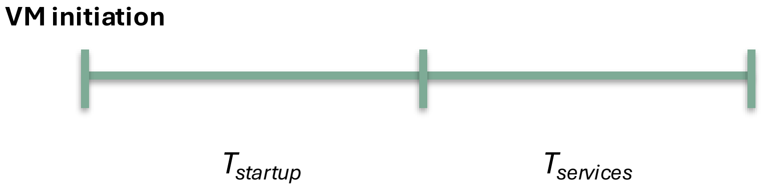
Shutdown time and restore time are generally negligible

Recovery process

$$T_{recovery} = \underbrace{T_{startup}}_{\text{Time to start VM}} + \underbrace{T_{services}}_{\text{Time to start Security Onion}}$$

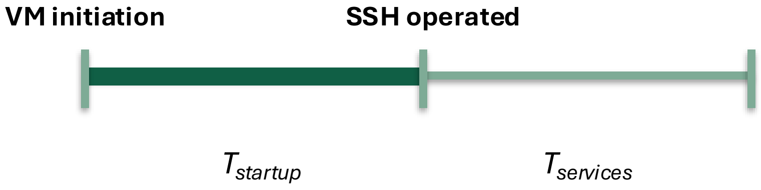
Recovery time includes VM startup and Security Onion service startup

Measurement Approach



$$T_{recovery} = T_{startup} + T_{services}$$

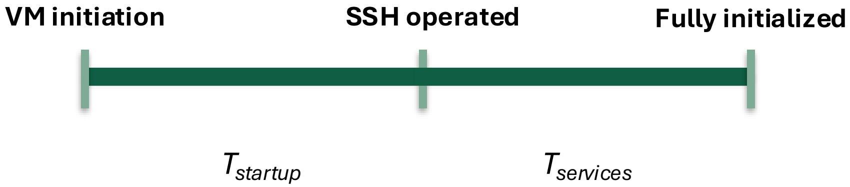
Measurement Approach



$$T_{recovery} = T_{startup} + T_{services}$$

Note: Using timestamp from system log

Measurement Approach



$$T_{recovery} = T_{startup} + T_{services}$$

Note: Using timestamp from system log

Scenarios

CPU Budget variation

I/O Bandwidth variation

Scenarios

CPU Budget variation

- To investigate impact of CPU contention
- CPU time is restricted within a period
- In units of percentage
- More CPU budget = Less CPU contention

I/O Bandwidth variation

Scenarios

CPU Budget variation

- To investigate impact of CPU contention
- CPU time is restricted within a period
- In units of percentage
- More CPU budget = Less CPU contention

I/O Bandwidth variation

- To investigate impact of I/O contention
- I/O throughput is restricted
- In units of IOPS
- More I/O bandwidth = Less I/O contention

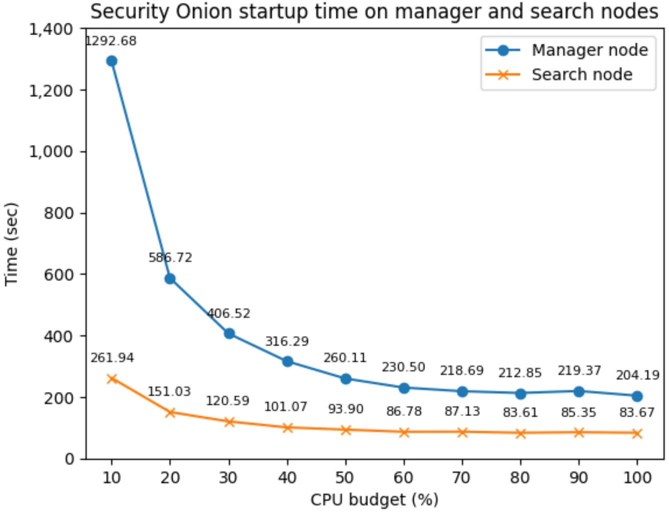
Results

Published paper

- **Title:** Effects of SIEM Recovery Time: Case Study on Security Onion
- **Published in:** 2024 21st International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON)
- **Date of Conference:** 27-30 May 2024
- **DOI:** 10.1109/ECTI-CON60892.2024.10594988

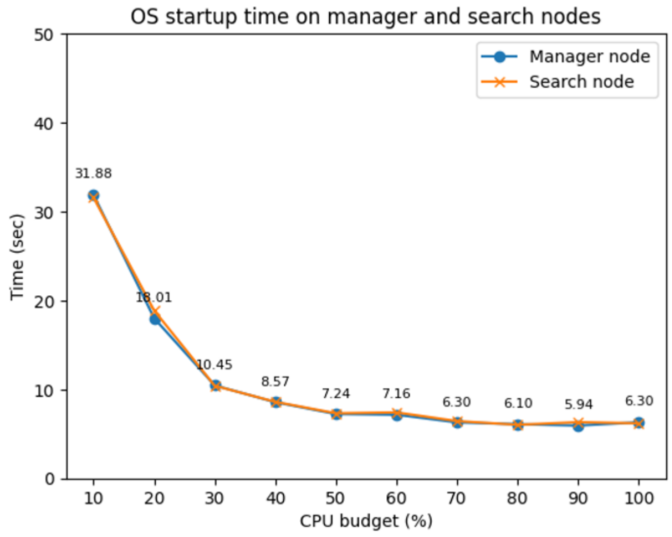
Influence of CPU Contention on Security Onion Startup Time

- Security Onion startup time for manager and search nodes under varying **CPU budgets**
- Exponential decay pattern for both nodes



Influence of CPU Contention on OS Startup Time

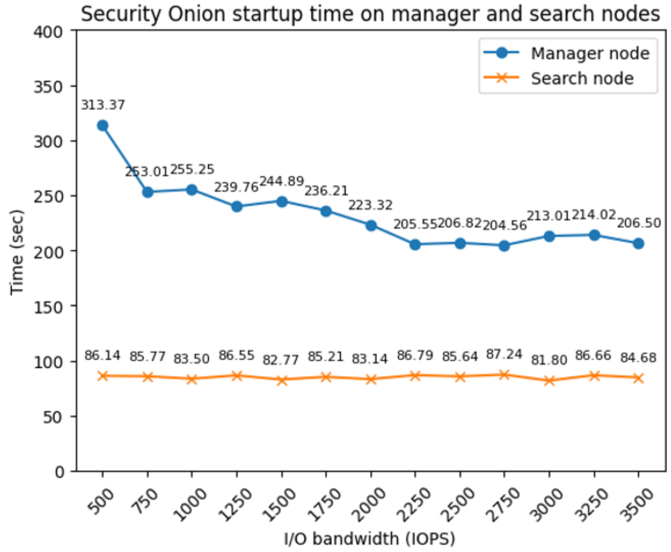
- OS startup time for manager and search nodes under varying **CPU budgets**
- Exponential decay pattern for both nodes



#HITB2024BKK

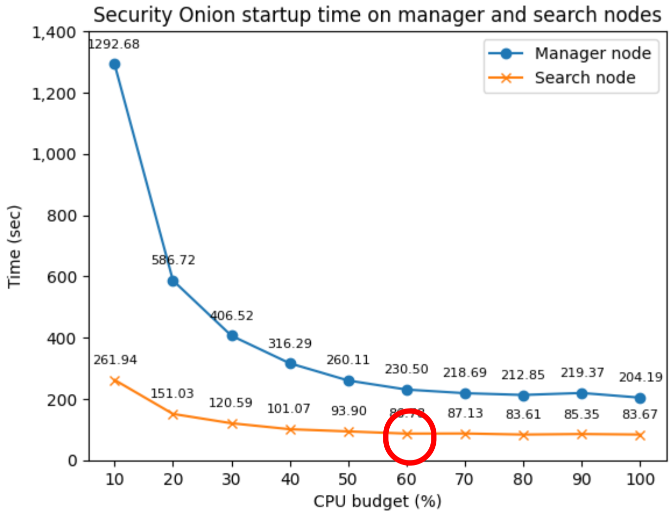
Influence of I/O Contention on Security Onion Startup Time

- Security Onion startup time for manager and search nodes under varying **I/O bandwidth**
- Linear for search node



Saturation point

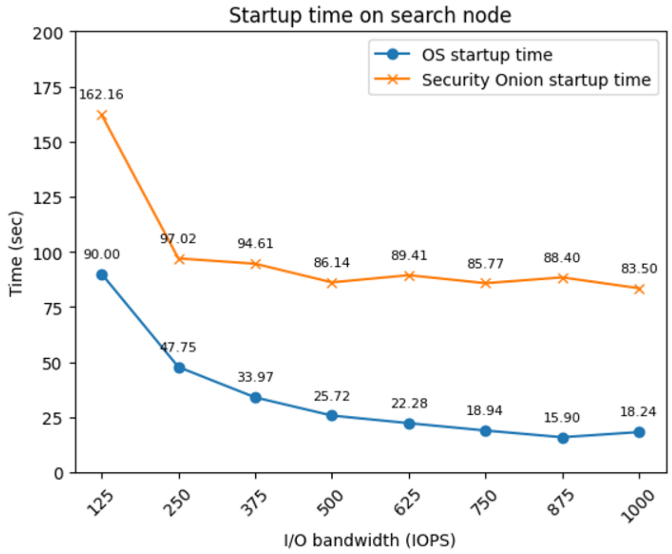
- Values that exceed the saturation point will no longer significantly affect to startup time



Saturation point

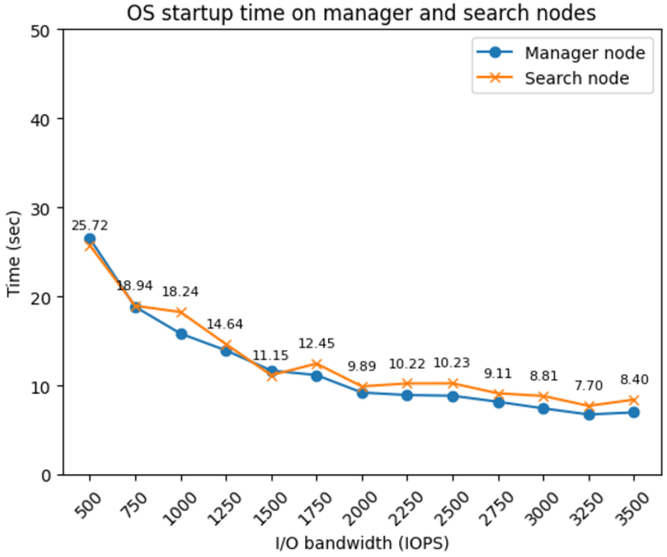
Influence of I/O Contention on Security Onion Startup Time

- Additional experiment for search node under varying lower **I/O bandwidth**
- Decreasing for both node



Influence of I/O Contention on OS Startup Time

- OS startup time for manager and search nodes under varying **I/O bandwidth**
- Exponential decay pattern for manager and search nodes



Summary

- Increased CPU and I/O contentions correlate with increased startup time
- Saturation points indicate threshold levels of resource requirements
- OS startup time consistent for both node types
- Security Onion startup time longer for manager nodes due to more complex service initiation
- Search nodes' I/O resource requirements lower than manager nodes

Key Takeaways

Multi-tenant Open-source SIEM

- **Challenge:** Open-source SIEMs often lack native multi-tenant capabilities
- **Solution:** Implemented a multi-tenant architecture with Security Onion
- **Key Features:**
  - Centralized authentication for shared sessions
  - Centralized permission management for simplified administration

Strategies for Enhancing Security Onion Recovery Time

- Allocate sufficient CPU and I/O resources, especially to manager nodes
- Consider CPU pinning to improve core utilization and reduce contention
- Utilize SSDs for faster I/O operations and quicker VM startup

Thank you!