# Assignment No – 5 (CSDF) (LP-IV)

**Code**

import time

class Log:

def \_\_init\_\_(self, timestamp, source, message):

self.timestamp = timestamp

self.source = source

self.message = message

class Event:

def \_\_init\_\_(self, category, severity, description):

self.category = category

self.severity = severity

self.description = description

class SIEM:

def \_\_init\_\_(self):

self.logs = []

self.events = []

def capture\_log(self, source, message):

timestamp = time.strftime('%Y-%m-%d %H:%M:%S')

log = Log(timestamp, source, message)

self.logs.append(log)

def correlate\_events(self):

for log in self.logs:

if "error" in log.message.lower():

event = Event("System Error", "High", f"Error detected from {log.source}: {log.message}")

self.events.append(event)

if "unauthorized" in log.message.lower():

event = Event("Unauthorized Access", "Medium", f"Unauthorized access attempt from {log.source}")

self.events.append(event)

def display\_events(self):

print("Events:")

for event in self.events:

print(f"Category: {event.category}\nSeverity: {event.severity}\nDescription: {event.description}\n")

if \_\_name\_\_ == "\_\_main\_\_":

siem = SIEM()

siem.capture\_log("Server-01", "Error: Disk space is critically low.")

siem.capture\_log("Firewall", "Unauthorized access attempt from IP address 192.168.1.100.")

siem.capture\_log("Server-02", "Application crashed unexpectedly.")

siem.capture\_log("Router", "Network traffic spiked significantly.")

siem.correlate\_events()

siem.display\_events()

**Output**

Events:

Category: System Error

Severity: High

Description: Error detected from Server-01: Error: Disk space is critically low.

Category: Unauthorized Access

Severity: Medium

Description: Unauthorized access attempt from Firewall: Unauthorized access attempt from IP address 192.168.1.100.