

Final Engagement

Attack, Defense & Analysis of a Vulnerable Network

Table of Contents

This document contains the following resources:



Alerts Implemented



Hardening



Implementing Patches





Alerts Implemented

Excessive HTTP Errors

Summarize the following:

- Which **metric** does this alert monitor?
 - `http.response.status_code`
- What is the **threshold** it fires at?
 - Top 5 Http status above 400 that happened in the last 5 minutes
- Provide a screenshot of the alert in action.

Edit Excessive HTTP Errors

Send an alert when your specified condition is met. Your watch will run every 1 minute.

Name
Excessive HTTP Errors

Indices to query
packetbeat-7.7.0 x

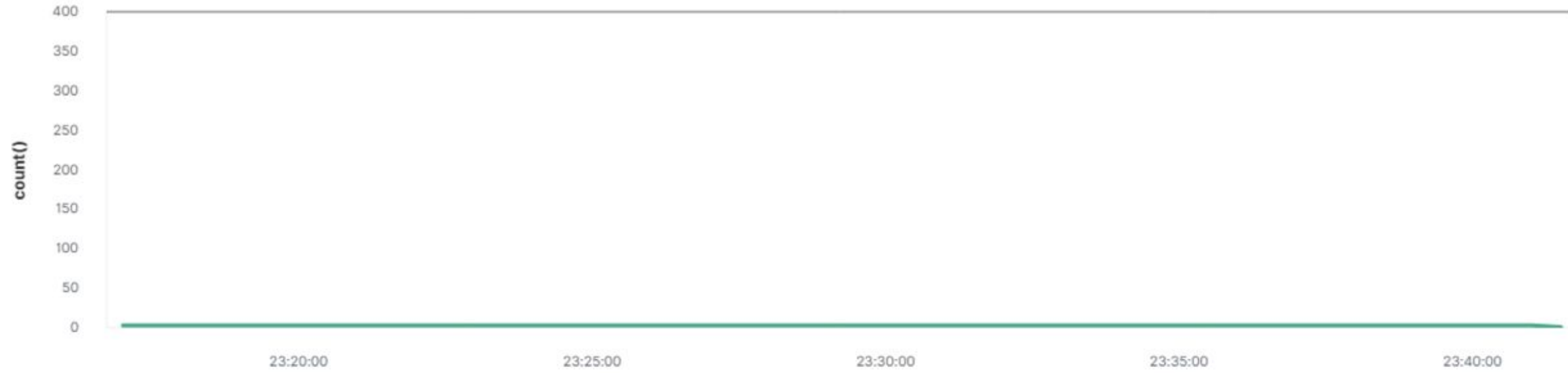
Time field
@timestamp

Run watch every
1 minute

Use * to broaden your query.

Match the following condition

WHEN count() GROUPED OVER top 5 'http.response.status_code' IS ABOVE 400 FOR THE LAST 5 minutes



Perform 0 actions when condition is met

Save alert Cancel

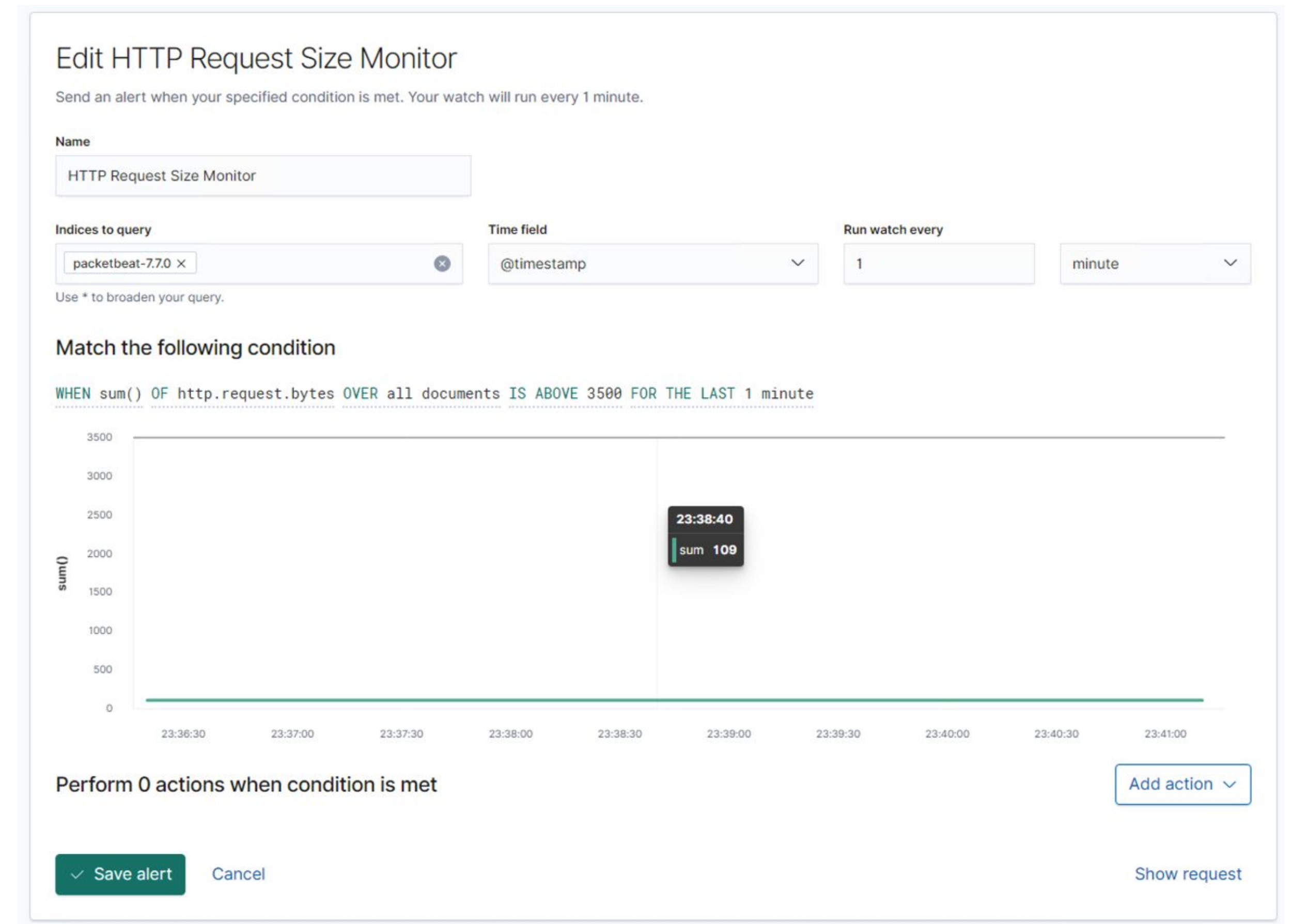
Add action

Show request

HTTP Request Size Monitor

Summarize the following:

- Which **metric** does this alert monitor?
 - `sum()` of `http.request.bytes`
- What is the **threshold** it fires at?
 - The sum of the `http.request.bytes` of all documents in the `http` request that above 3000 for the last 1 minute.
- Provide a screenshot of the alert in action.



CPU Usage Monitor

Summarize the following:

- Which **metric** does this alert monitor?
 - max() OF system.process.cpu.total.pct
- What is the **threshold** it fires at?
 - The max of the system.process.cpu.total.pct of all document is above 50% over the last 5 minutes
- Provide a screenshot of the alert in action.

Create threshold alert

Send an alert when your specified condition is met. Your watch will run every 1 minute.

Name
CPU Usage Monitor

Indices to query
metricbeat-7.7.0 x

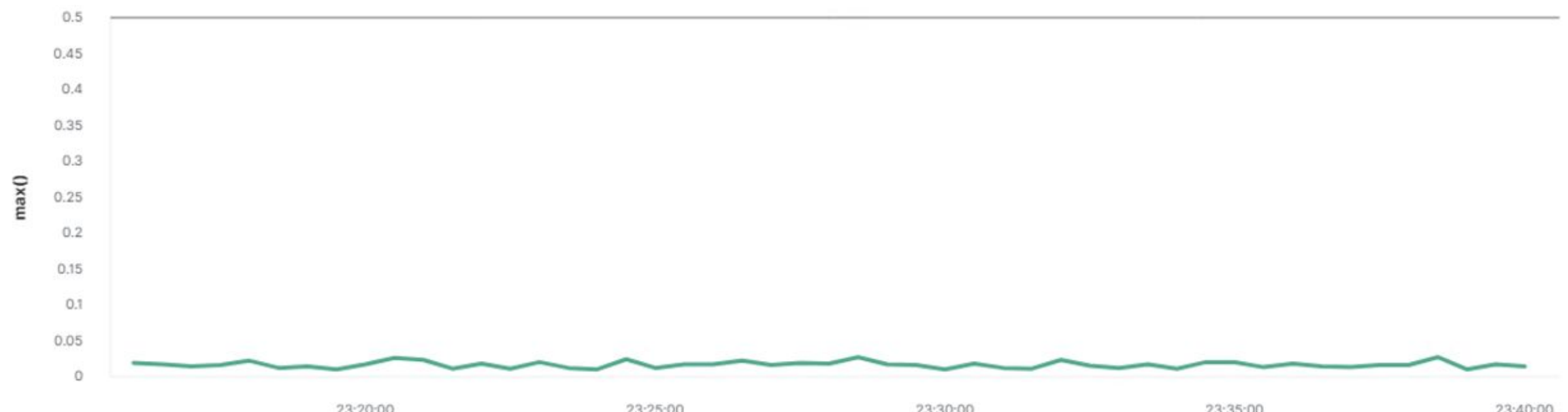
Time field
@timestamp

Run watch every
1 minute

Use * to broaden your query.

Match the following condition

WHEN max() OF system.process.cpu.total.pct OVER all documents IS ABOVE 0.5 FOR THE LAST 5 minutes



Perform 0 actions when condition is met

Add action

Create alert Cancel Show request

Hardening

Mitigations of Vulnerabilities

- **Weak Password - Better password policies**
 - Why the patch works.
 - Stronger, harder to guess passwords
- **Privilege Escalation due to misconfiguration - Require root password when Steven uses python as root, or remove his privilege to run python as root**
 - Why the patch works.
 - Users should not be given permission to run applications as root without the root password
 - How to install it (include commands)
 - Use `visudo` to change the follow line for Steven:
steven ALL=(ALL):/usr/bin/python # or remove it entirely
- **Vulnerable and Outdated Components**
 - Why the patch works.
 - Install software updates to patch for known vulnerabilities.
 - How to install it (include commands)
 - See slide for “Implementing Patches with Ansible”

Implementing Patches

Implementing Patches with Ansible

Vulnerability - Vulnerable and Outdated Components

Update the software to the latest version to avoid any known vulnerabilities.

Playbook Overview

- name: Update all the installed packages

hosts: target1

become: true

tasks:

- name: apt update

- command: "apt update"

- name: apt upgrade

- command: "apt upgrade"



The End