Final Engagement

Attack, Defense & Analysis of a Vulnerable Network

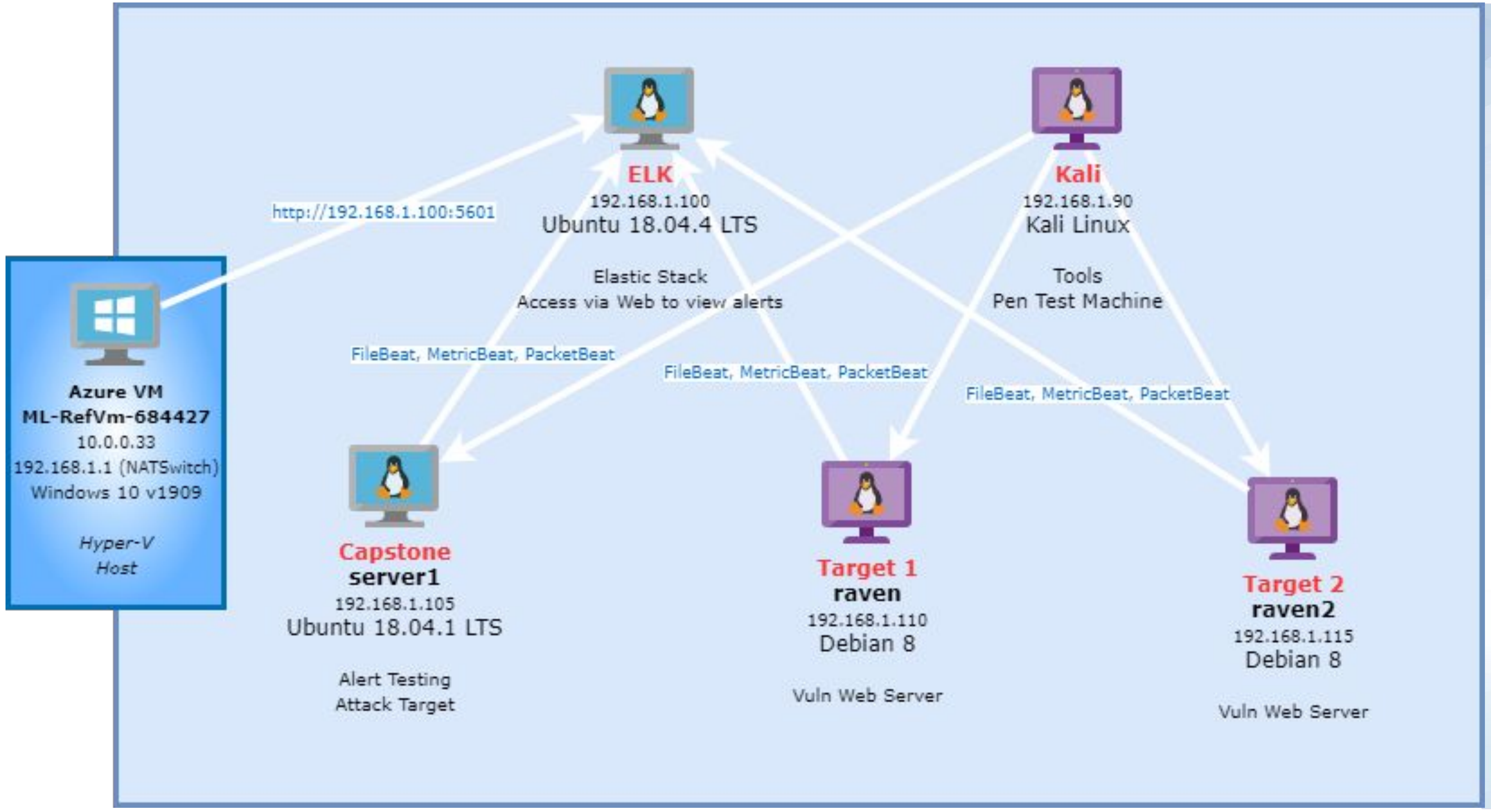
Asha, Sam, David, Phil, Pritesh

Table of Contents

This document contains the following resources:

Network Topology
& Critical Vulnerabilities

Network Topology



Network
Address Range:
192.168.1.0/24
Netmask: 255.255.255.0
Gateway:192.168.1.1

Machines
IPv4: 192.168.1.90
OS: Kali Linux Rolling
Hostname: Kali Machine

IPv4: 192.168.1.100
OS: Ubuntu 18.04.4 LTS
Hostname: ELK

IPv4: 192.168.1.105
OS: Ubuntu 18.04.1 LTS
Hostname: Capstone

IPv4: 192.168.1.110
OS: Debian 8 (jessie)
Hostname: Target 1

Critical Vulnerabilities: Target 1

Our assessment uncovered the following critical vulnerabilities in **Target 1**.

Vulnerability	Description	Impact
CWE-200: Exposure of Sensitive Information to an Unauthorized Actor **Port Scanning** Src: https://cwe.mitre.org/data/definitions/200.html	The product exposes sensitive information to an actor that is not explicitly authorized to have access to that information. Some kinds of sensitive information include: private, personal information, such as personal messages, financial data, health records, geographic location, or contact details, system status and environment, such as the operating system and installed packages, business secrets and intellectual property, network status and configuration, the product's own code or internal state, metadata, e.g. logging of connections or message headers, indirect information, such as a discrepancy between two internal operations that can be observed by an outsider.	Port scanning allows a malicious actor to identify where the targets weakness is for entry, and list potential exploits that can be used.
CWE-521: Weak Password Requirements Src: https://www.cvedetails.com/cwe-details/521/Weak-Password-Requirements.html	Authentication mechanisms often rely on a memorized secret (also known as a password) to provide an assertion of identity for a user of a system. It is therefore important that this password be of sufficient complexity and impractical for an adversary to guess.	Passwords that are easily guessed can lead to unauthorised access into a system or network and expose sensitive data.
CWE-284: Improper Access Control Src: https://www.cvedetails.com/cwe-details/284/Access-Control-Authorization-Issues.html	The software does not restrict or incorrectly restricts access to a resource from an unauthorized actor. Access control involves the use of several protection mechanisms such as authentication, authorization and accountability.	When any mechanism is not applied or otherwise fails, attackers can compromise the security of the software by gaining privileges, reading sensitive information, executing commands, evading detection, etc.

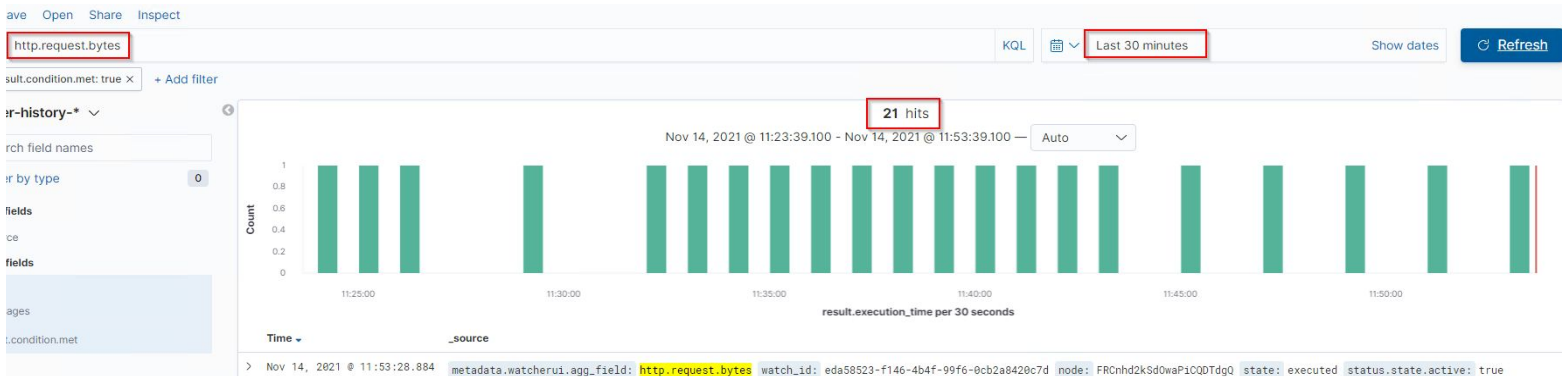
Alerts Implemented

HTTP Request Size Monitor

Summarize the following:

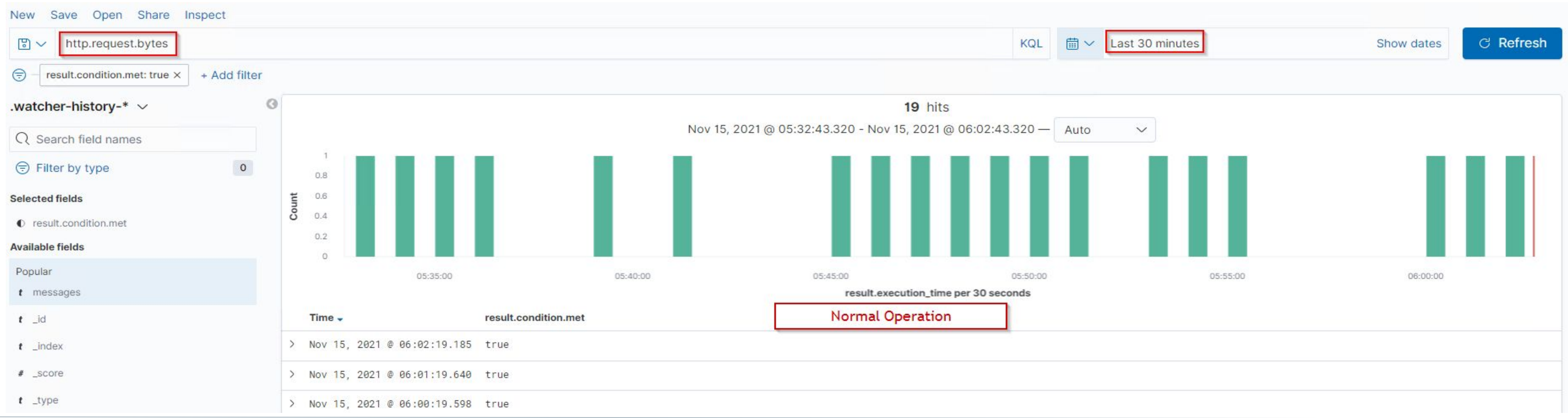
- Which **metric** does this alert monitor? http.request.bytes
- What is the **threshold** it fires at? total of 3,500 bytes in the last minute

Attack Hits



Time	_source
Nov 14, 2021 @ 11:53:28.884	metadata.watcherui.agg_field: http.request.bytes watch_id: eda58523-f146-4b4f-99f6-0cb2a8420c7d node: FRCnhd2kSdOwaPiCQDTdgQ state: executed status.state.active: true

Normal Operation

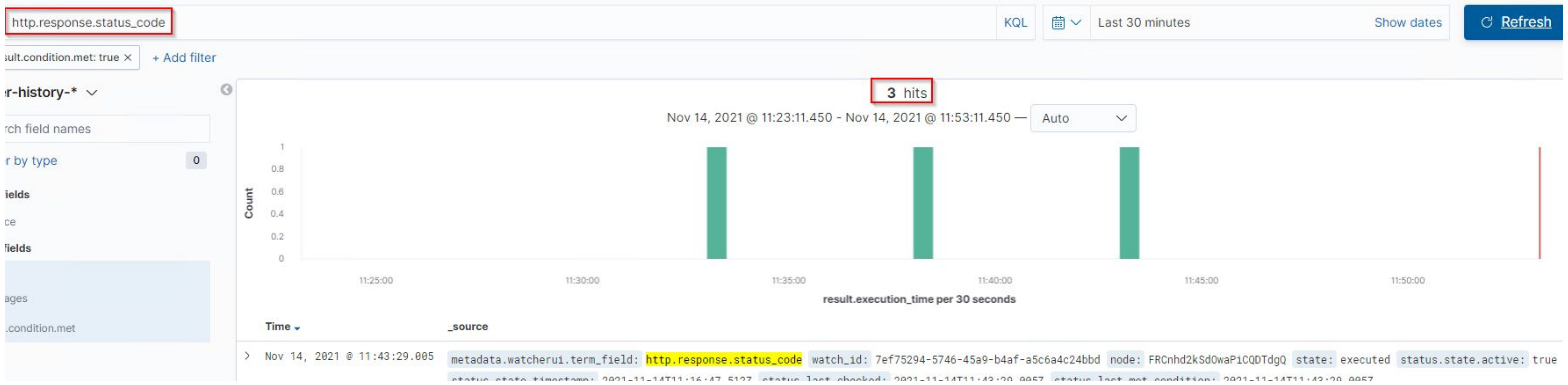


Time	result.condition.met
Nov 15, 2021 @ 06:02:19.185	true
Nov 15, 2021 @ 06:01:19.640	true
Nov 15, 2021 @ 06:00:19.598	true

Excessive HTTP Errors

Summarize the following:

- Which **metric** does this alert monitor? http.response.status_code
- What is the **threshold** it fires at? Above 400 for the last 5 minutes

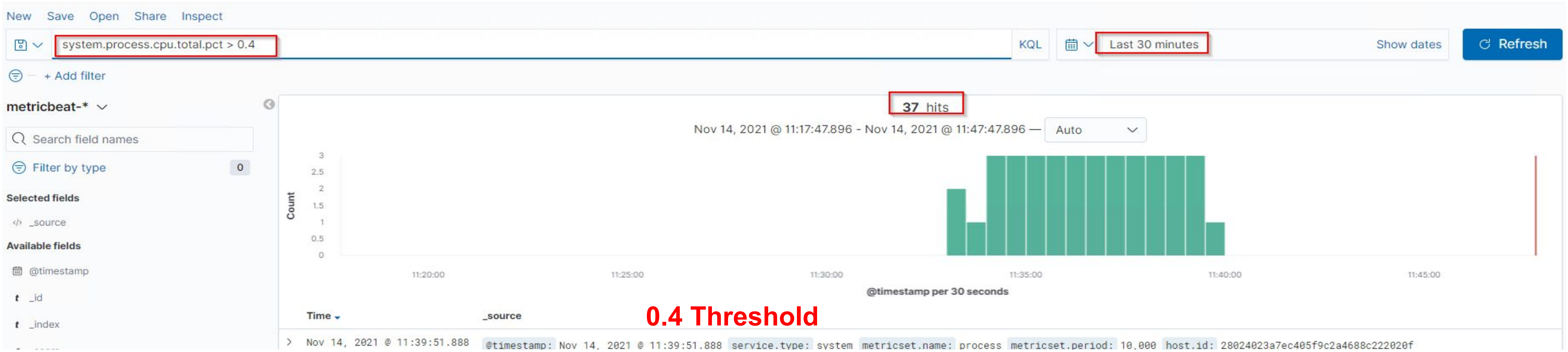


```
Nov 14, 2021 @ 11:43:29.005   metadata.watcherui.term_field: http.response.status_code watch_id: 7ef75294-5746-45a9-b4af-a5c6a4c24bbd node: FRCnhd2kSdOwaPiCQDTdgQ state: executed status.state.active: true
```

CPU Usage Monitor

Summarize the following:

- Which **metric** does this alert monitor? system.process.cpu.total.pct
- What is the **threshold** it fires at? Above 0.5 for the last 5 minutes (0.5 equates to 50% of the non-idle time and non I/O wait time)



system.process.cpu.total.pct > 0.4

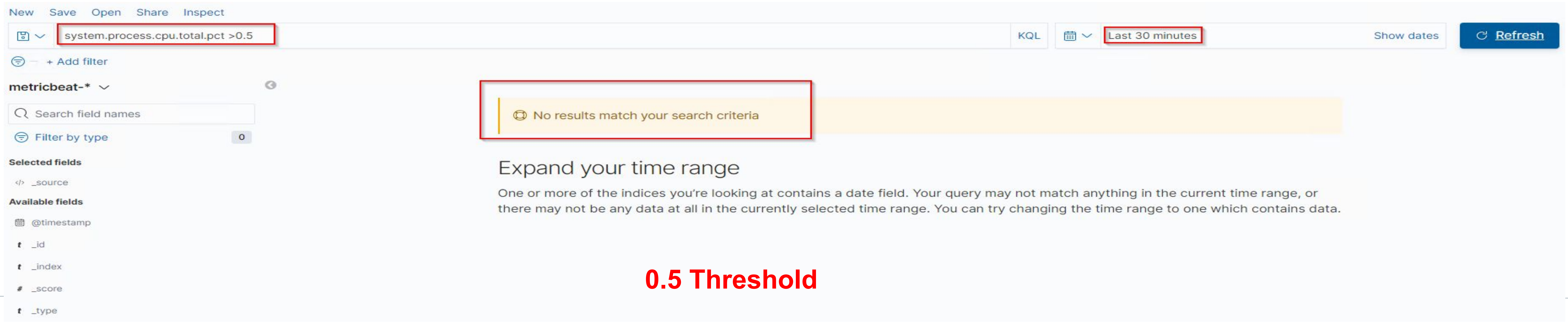
Last 30 minutes

37 hits

Nov 14, 2021 @ 11:17:47.896 - Nov 14, 2021 @ 11:47:47.896

```
Nov 14, 2021 @ 11:39:51.888   @timestamp: Nov 14, 2021 @ 11:39:51.888 service.type: system metricset.name: process metricset.period: 10,000 host.id: 28024023a7ec405f9c2a4688c222020f
```

0.4 Threshold



system.process.cpu.total.pct >0.5

Last 30 minutes

No results match your search criteria

Expand your time range

One or more of the indices you're looking at contains a date field. Your query may not match anything in the current time range, or there may not be any data at all in the currently selected time range. You can try changing the time range to one which contains data.

0.5 Threshold

Hardening

Hardening Against Weak Authentication on Target 1

PASSWORD POLICY:

- Create a strong, Long Passphrase including the use of upper/lowercase, special characters and numeric characters.
- Mandatory periodic password change
- Passwords cannot contain the username or any identifiable details of the user.
- Enforce a password history policy to eliminate the repeated use of a password.
- Suspend Accounts after 45 days without a valid login
- Limit failed Login attempts
- Monitor Failed Login Attempts

ADDITIONAL LAYERS OF AUTHENTICATION:

- Implement Multi-Factor Authentication
- RSA Hard Token
- CAPTCHA tools to differentiate between real users and automated users
- Use of Password Managers (LastPass)

Account lockout/Authentication Policy hardening.

- This will make it more difficult for an intruder to circumvent the authentication process.
- Implementation of the Password Policy includes but is not to limited to the following;

To set the minimum password length:

```
sudo nano /etc/pam.d/common-password
password [success=2 default=ignore] pam_unix.so obscure sha512 minlen=12
```

To enforce complexity of passwords:

```
sudo apt-get install libpam-pwquality
sudo nano /etc/pam.d/common-password
```

To set 1 character as uppercase:

```
password    requisite   pam_pwquality.so retry=3 ucredit=-1
```

To set expiry dates:

```
sudo nano /etc/login.defs
PASS_MAX_DAYS 90
PASS_MIN_DAYS   0
PASS_WARN_AGE  7
```

Hardening Against SSH authentication Vulnerability on Target 1

Implement SSH only via SSH-key and not password

SSH-key will only allow administrators with the SSH-key access to the server

- Command to disable SSH via password

  sudo vi /etc/ssh/sshd_config

then set….

ChallengeResponseAuthentication to no
PasswordAuthentication to no
UsePAM to no
PermitRootLogin to no

- Setting up firewall rule - allow only SSH from a list of whitelisted IP addresses
- The firewall rule will only allow access from the administrators IP address
- Command to set up firewall to allow whitelisted IP address

  sudo ufw allow from [admin’s IP address] proto tcp to any port 22

Hardening Against folder enumeration Vulnerability on Target 1

With the implementation of POLP (Principle of Least Privilege) the access to the folders would be restricted and by which Michael wouldn't be able to access wp-config.php where database username and password were found.

- Why the patch works: It reduces the attack surface, and only allows required access, it would have eliminated the exposure of mysql root credentials.
- How to install it (include commands).

```
- chmod 400 [filename/foldername]
- chmod 400 wp-config.php
```

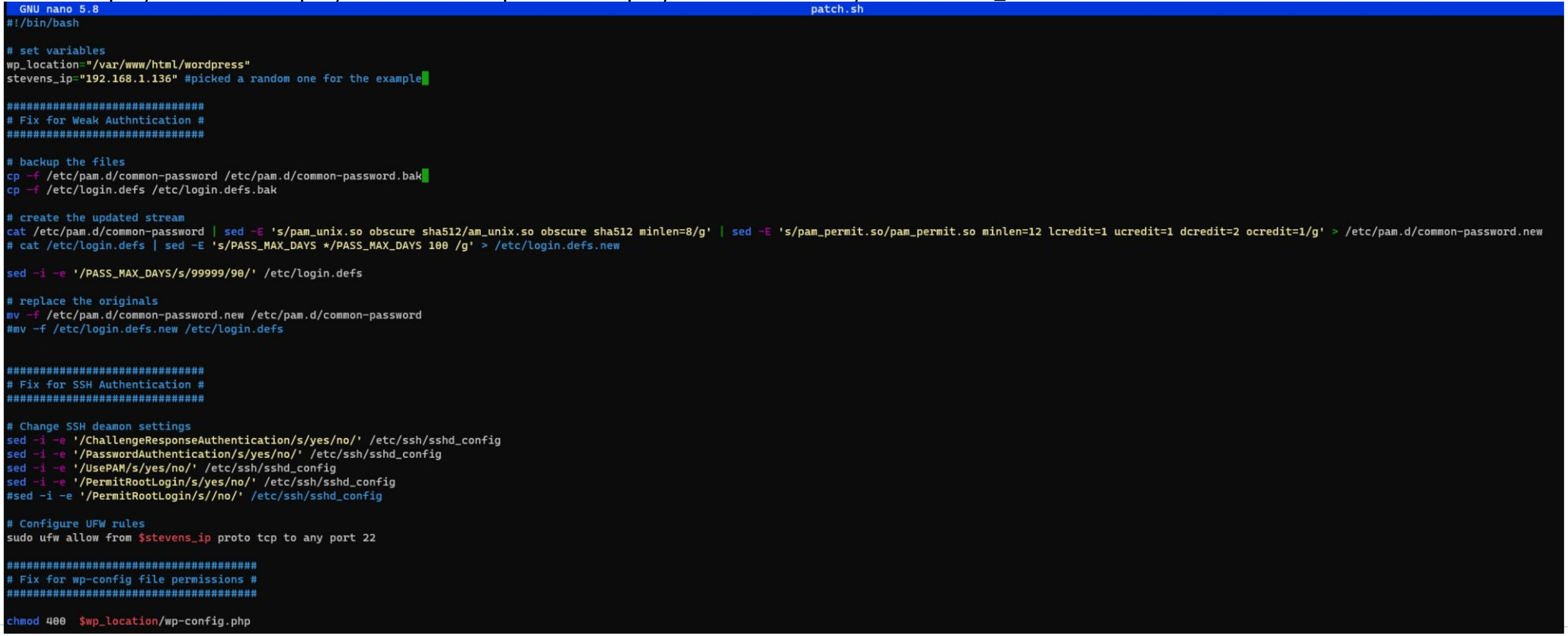
Implementing Patches

Implementing Patches with Ansible

Playbook Overview

- name: Run a script to patch(free form)

- **ansible.builtin.script:** /patch.sh arguments
- Copy the vulnerable.yml file to /etc/ansible/files.
- Update the /etc/ansible/hosts file to include Target1(192.168.1.110) and Target2(192.168.1.115)
- Run the playbook ansible-playbook for example ansible-playbook vulnerable.yml -i ansible_hosts



```
  GNU nano 5.8                                                    patch.sh
#!/bin/bash

# set variables
wp_location="/var/www/html/wordpress"
stevens_ip="192.168.1.136" #picked a random one for the example

###############################
# Fix for Weak Authntication #
###############################

# backup the files
cp -f /etc/pam.d/common-password /etc/pam.d/common-password.bak
cp -f /etc/login.defs /etc/login.defs.bak

# create the updated stream
cat /etc/pam.d/common-password | sed -E 's/pam_unix.so obscure sha512/am_unix.so obscure sha512 minlen=8/g' | sed -E 's/pam_permit.so/pam_permit.so minlen=12 lcredit=1 ucredit=1 dcredit=2 ocredit=1/g' > /etc/pam.d/common-password.new
# cat /etc/login.defs | sed -E 's/PASS_MAX_DAYS */PASS_MAX_DAYS 100 /g' > /etc/login.defs.new

sed -i -e '/PASS_MAX_DAYS/s/99999/90/' /etc/login.defs

# replace the originals
mv -f /etc/pam.d/common-password.new /etc/pam.d/common-password
#mv -f /etc/login.defs.new /etc/login.defs


###############################
# Fix for SSH Authentication #
###############################

# Change SSH deamon settings
sed -i -e '/ChallengeResponseAuthentication/s/yes/no/' /etc/ssh/sshd_config
sed -i -e '/PasswordAuthentication/s/yes/no/' /etc/ssh/sshd_config
sed -i -e '/UsePAM/s/yes/no/' /etc/ssh/sshd_config
sed -i -e '/PermitRootLogin/s/yes/no/' /etc/ssh/sshd_config
#sed -i -e '/PermitRootLogin/s//no/' /etc/ssh/sshd_config

# Configure UFW rules
sudo ufw allow from $stevens_ip proto tcp to any port 22

######################################
# Fix for wp-config file permissions #
######################################

chmod 400  $wp_location/wp-config.php
```