Final Engagement

Attack, Defense & Analysis of a Vulnerable Network

Iris Carrell, Carolina Hernandez, Crystal Hamilton, Jacob Starks, Braden Welsh

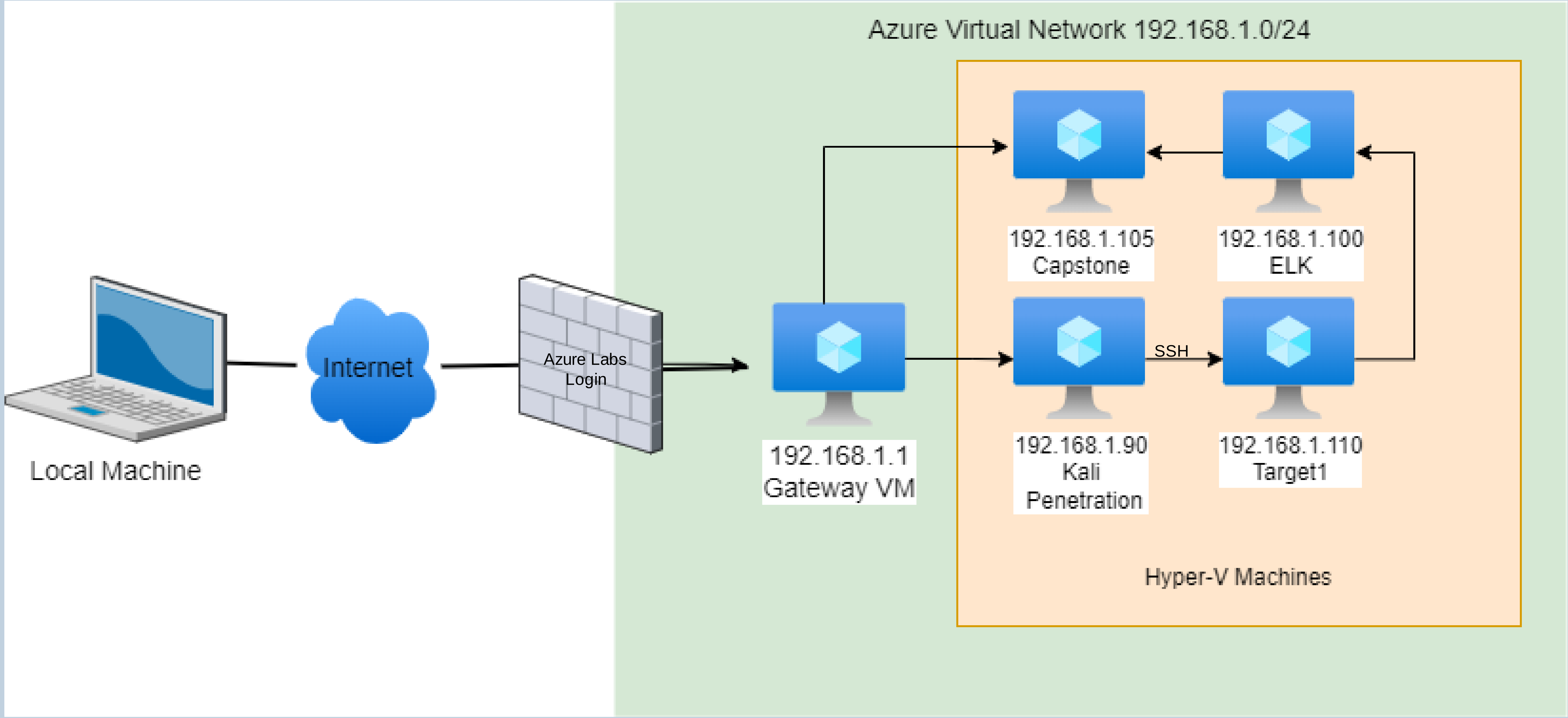
Table of Contents

This document contains the following resources:

- **Network Topology & Critical Vulnerabilities**
- **Alerts Implemented**
- **Hardening**
- **Implementing Patches**

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 5.4.0
Hostname: Kali

IPv4: 192.168.1.110
OS: Linux
Hostname: Target 1
Purpose: Wordpress

IPv4: 192.168.1.100
OS: Linux
Hostname: Elk

IPv4: 192.168.1.105
OS: Linux Ubuntu
Hostname: Capstone

Critical Vulnerabilities: Target 1

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

Vulnerability	Description	Impact
Vulnerable Ports 22 and 80	Access to machine via OpenSSH, Scans and direct access to the Target 1 machine	Integrity and confidentiality because of direct access to machine and ability to gain more details about users/visitors
Weak/Insecure Passwords	The user Michael has a guessable password which could also be cracked via brute force methods	Integrity and Confidentiality due to the ability to breach the machine and gain more information about users/operations
Enumerate WordPress Site	Users were identifiable via WPScan	Confidentiality is impacted through the disclosure of usernames and other details

Critical Vulnerabilities: Target 1

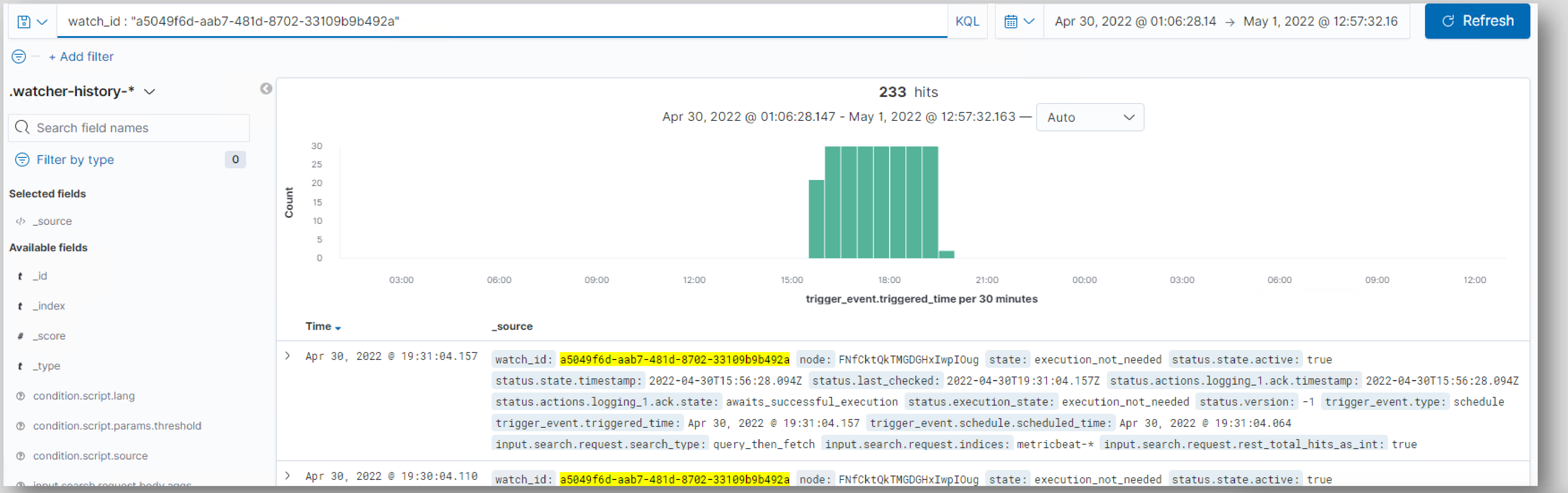
Our assessment uncovered the following critical vulnerabilities in **Target 1**.

Vulnerability	Description	Impact
Apache 2.4.10 CVE-2016-4975	Apache Server can be vulnerable for CRLF Injection	Integrity impact as it allows the attacker to set fake cookies, steal CSRF tokens, disclose user information by injecting a script (XSS) and perform a variety of other attacks. It also allows attackers to deactivate & bypass security measures like XSS filters & Same Origin Policy (SOP) (See more at (CRLF Injection Attack - (https://www.geeksforgeeks.org/crlf-injection-attack/))
Python Privilege Escalation	The user Steven can circumvent lower privileges by using python scripting allowed for sudo	Integrity and Confidentiality by gaining root access to the machine

Alerts Implemented

HTTP Request Size Monitor

- This monitoring rule watches the http.request.bytes from metricbeat
- It will fire when it exceeds a sum of 3500 (3.5mb) for the last minute
- The condition syntax is WHEN sum() OF http.request.bytes OVER all documents IS ABOVE 3500 FOR THE LAST 1 minute



watch_id : "a5049f6d-aab7-481d-8702-33109b9b492a"

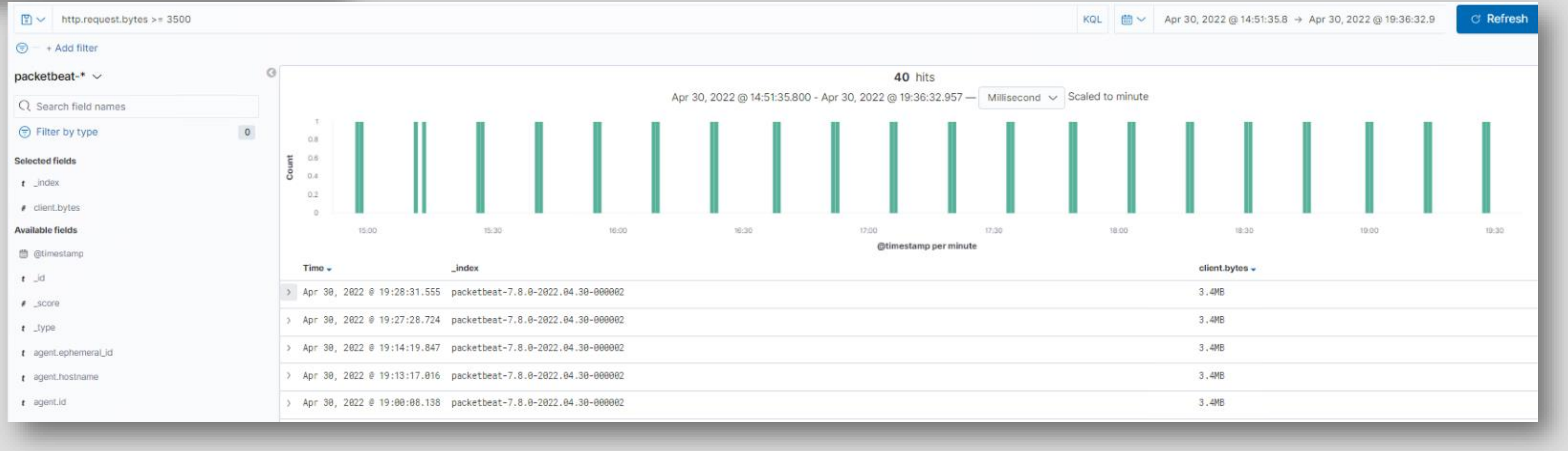
Apr 30, 2022 @ 01:06:28.14 → May 1, 2022 @ 12:57:32.16

.watcher-history-*

233 hits

Apr 30, 2022 @ 01:06:28.147 - May 1, 2022 @ 12:57:32.163 — Auto

```
> Apr 30, 2022 @ 19:31:04.157  watch_id: a5049f6d-aab7-481d-8702-33109b9b492a node: FNfCktQkTMGDGHxIwpIOug state: execution_not_needed status.state.active: true
                               status.state.timestamp: 2022-04-30T15:56:28.094Z status.last_checked: 2022-04-30T19:31:04.157Z status.actions.logging_1.ack.timestamp: 2022-04-30T15:56:28.094Z
                               status.actions.logging_1.ack.state: awaits_successful_execution status.execution_state: execution_not_needed status.version: -1 trigger_event.type: schedule
                               trigger_event.triggered_time: Apr 30, 2022 @ 19:31:04.157 trigger_event.schedule.scheduled_time: Apr 30, 2022 @ 19:31:04.064
                               input.search.request.search_type: query_then_fetch input.search.request.indices: metricbeat-* input.search.request.rest_total_hits_as_int: true
> Apr 30, 2022 @ 19:30:04.110  watch_id: a5049f6d-aab7-481d-8702-33109b9b492a node: FNfCktQkTMGDGHxIwpIOug state: execution_not_needed status.state.active: true
```



http.request.bytes >= 3500

Apr 30, 2022 @ 14:51:35.8 → Apr 30, 2022 @ 19:36:32.9

packetbeat-*

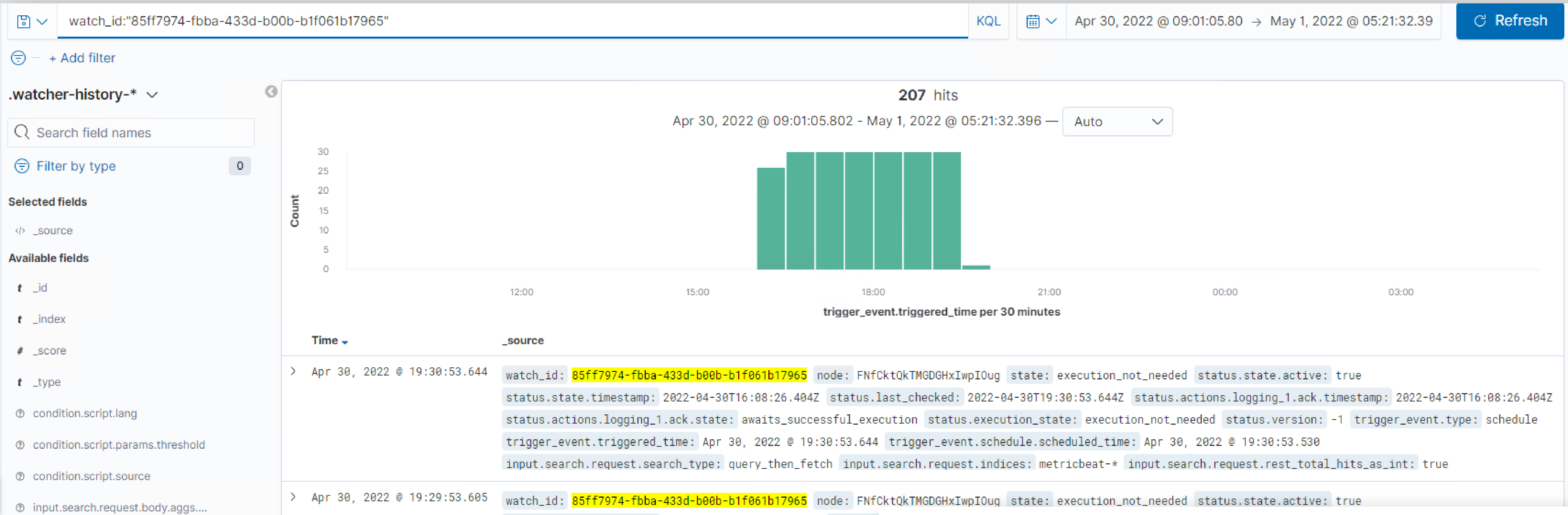
40 hits

Apr 30, 2022 @ 14:51:35.800 - Apr 30, 2022 @ 19:36:32.957 — Millisecond Scaled to minute

```
  Time                          _index                                      client.bytes
> Apr 30, 2022 @ 19:28:31.555  packetbeat-7.8.0-2022.04.30-000002           3.4MB
> Apr 30, 2022 @ 19:27:28.724  packetbeat-7.8.0-2022.04.30-000002           3.4MB
> Apr 30, 2022 @ 19:14:19.847  packetbeat-7.8.0-2022.04.30-000002           3.4MB
> Apr 30, 2022 @ 19:13:17.016  packetbeat-7.8.0-2022.04.30-000002           3.4MB
> Apr 30, 2022 @ 19:00:08.138  packetbeat-7.8.0-2022.04.30-000002           3.4MB
```

Excessive HTTP Errors

- This monitoring rule watches the http.response.status_code from metricbeat
- It will fire when it reaches above a count of 400 for the last 5 minutes
- The condition syntax is WHEN count() GROUPED OVER top 5 'http.response.status_code' IS ABOVE 400 FOR THE LAST 5 minutes



watch_id:"85ff7974-fbba-433d-b00b-b1f061b17965"

Apr 30, 2022 @ 09:01:05.80 → May 1, 2022 @ 05:21:32.39

.watcher-history-*

Search field names

Filter by type 0

Selected fields

- _source

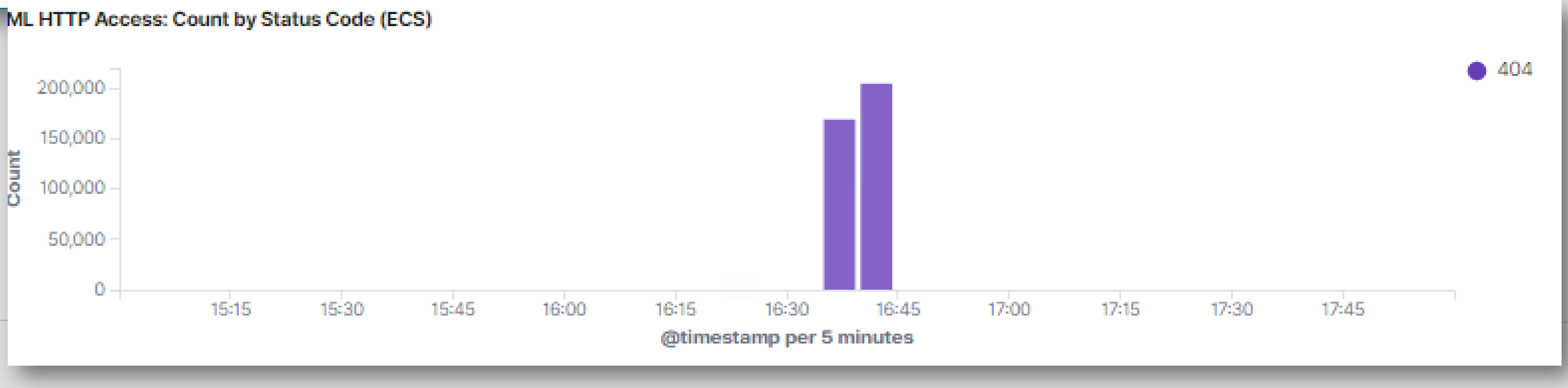
Available fields

- _id
- _index
- _score
- _type
- condition.script.lang
- condition.script.params.threshold
- condition.script.source
- input.search.request.body.aggs....

207 hits

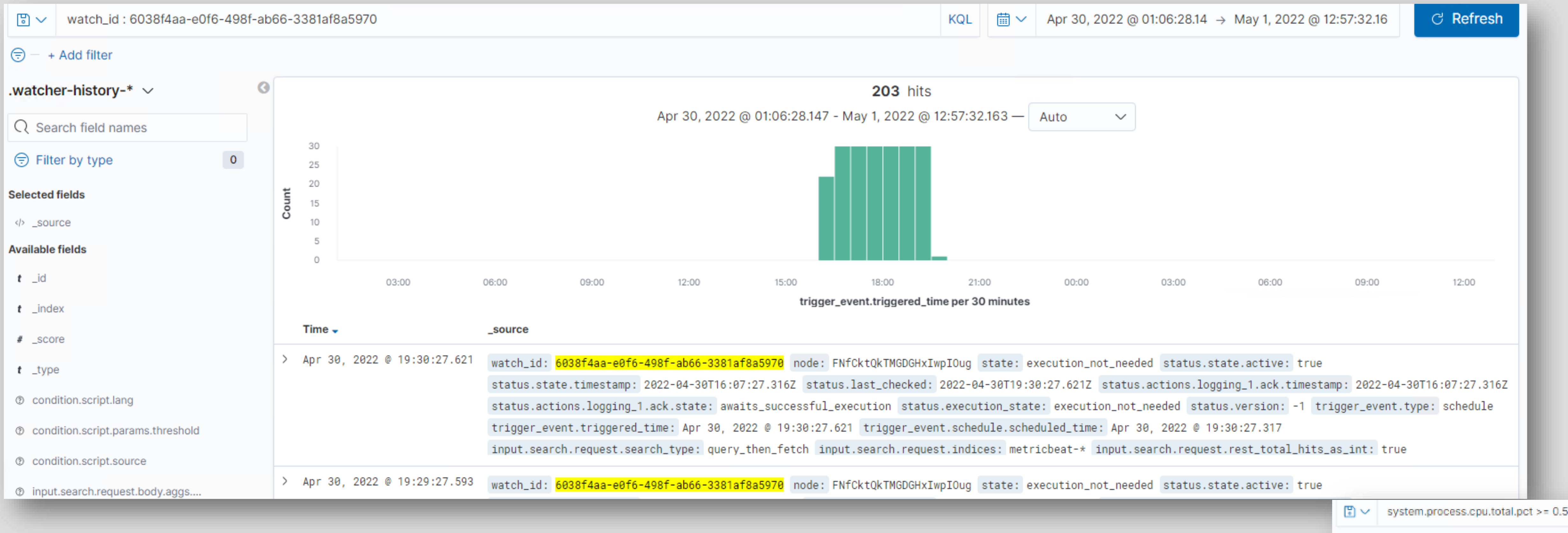
Apr 30, 2022 @ 09:01:05.802 - May 1, 2022 @ 05:21:32.396 —

```
Time      _source
Apr 30, 2022 @ 19:30:53.644   watch_id: 85ff7974-fbba-433d-b00b-b1f061b17965 node: FNfCktQkTMGDGHxIwpIOug state: execution_not_needed status.state.active: true
                              status.state.timestamp: 2022-04-30T16:08:26.404Z status.last_checked: 2022-04-30T19:30:53.644Z status.actions.logging_1.ack.timestamp: 2022-04-30T16:08:26.404Z
                              status.actions.logging_1.ack.state: awaits_successful_execution status.execution_state: execution_not_needed status.version: -1 trigger_event.type: schedule
                              trigger_event.triggered_time: Apr 30, 2022 @ 19:30:53.644 trigger_event.schedule.scheduled_time: Apr 30, 2022 @ 19:30:53.530
                              input.search.request.search_type: query_then_fetch input.search.request.indices: metricbeat-* input.search.request.rest_total_hits_as_int: true
Apr 30, 2022 @ 19:29:53.605   watch_id: 85ff7974-fbba-433d-b00b-b1f061b17965 node: FNfCktQkTMGDGHxIwpIOug state: execution_not_needed status.state.active: true
```

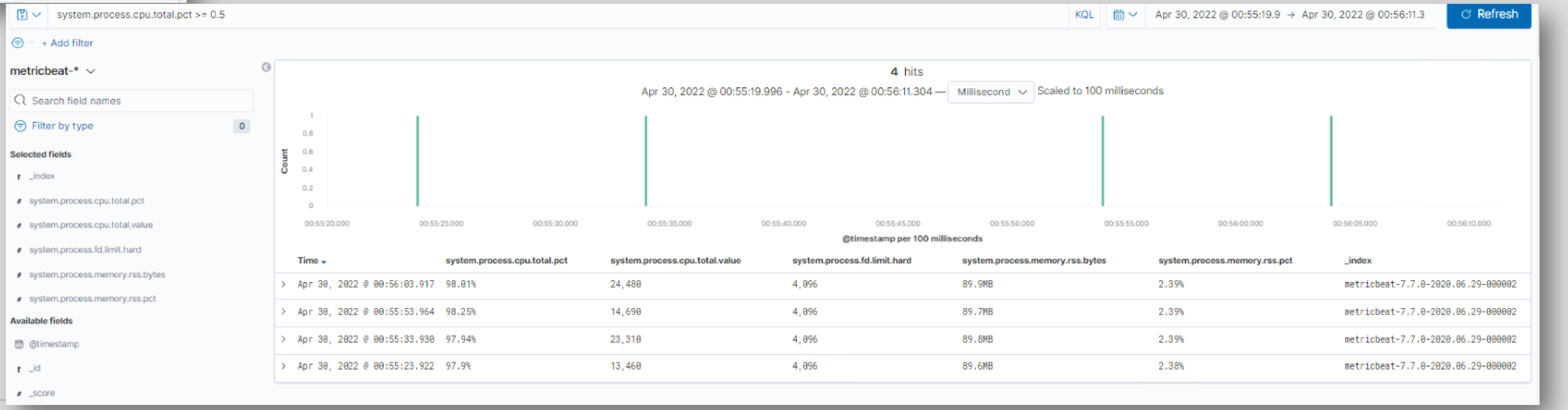


CPU Usage Monitor

- This monitoring rule watches the system.process.cpu.total.pct from metricbeat
- It will fire when its max value remains above 0.5 over all processes for the last 5 minutes
- The condition syntax is WHEN max() OF system.process.cpu.total.pct OVER all documents IS ABOVE 0.5 FOR THE LAST 5 minutes



```
> Apr 30, 2022 @ 19:30:27.621   watch_id: 6038f4aa-e0f6-498f-ab66-3381af8a5970 node: FNfCktQkTMGDGHxIwpIOug state: execution_not_needed status.state.active: true
                                status.state.timestamp: 2022-04-30T16:07:27.316Z status.last_checked: 2022-04-30T19:30:27.621Z status.actions.logging_1.ack.timestamp: 2022-04-30T16:07:27.316Z
                                status.actions.logging_1.ack.state: awaits_successful_execution status.execution_state: execution_not_needed status.version: -1 trigger_event.type: schedule
                                trigger_event.triggered_time: Apr 30, 2022 @ 19:30:27.621 trigger_event.schedule.scheduled_time: Apr 30, 2022 @ 19:30:27.317
                                input.search.request.search_type: query_then_fetch input.search.request.indices: metricbeat-* input.search.request.rest_total_hits_as_int: true
> Apr 30, 2022 @ 19:29:27.593   watch_id: 6038f4aa-e0f6-498f-ab66-3381af8a5970 node: FNfCktQkTMGDGHxIwpIOug state: execution_not_needed status.state.active: true
```


system.process.cpu.total.pct >= 0.5
KQL
Apr 30, 2022 @ 00:55:19.9 → Apr 30, 2022 @ 00:56:11.3
Refresh
+ Add filter
metricbeat-*
Search field names
Filter by type
0
Selected fields
_index
system.process.cpu.total.pct
system.process.cpu.total.value
system.process.fd.limit.hard
system.process.memory.rss.bytes
system.process.memory.rss.pct
Available fields
@timestamp
_id
_score
4 hits
Apr 30, 2022 @ 00:55:19.996 - Apr 30, 2022 @ 00:56:11.304 —
Millisecond
Scaled to 100 milliseconds
Count
1
0.8
0.6
0.4
0.2
0
00:55:20.000
00:55:25.000
00:55:30.000
00:55:35.000
00:55:40.000
00:55:45.000
00:55:50.000
00:55:55.000
00:56:00.000
00:56:05.000
00:56:10.000
@timestamp per 100 milliseconds

Time	system.process.cpu.total.pct	system.process.cpu.total.value	system.process.fd.limit.hard	system.process.memory.rss.bytes	system.process.memory.rss.pct	_index
Apr 30, 2022 @ 00:56:03.917	98.01%	24,480	4,096	89.9MB	2.39%	metricbeat-7.7.0-2020.06.29-000002
Apr 30, 2022 @ 00:55:53.964	98.25%	14,690	4,096	89.7MB	2.39%	metricbeat-7.7.0-2020.06.29-000002
Apr 30, 2022 @ 00:55:33.930	97.94%	23,310	4,096	89.8MB	2.39%	metricbeat-7.7.0-2020.06.29-000002
Apr 30, 2022 @ 00:55:23.922	97.9%	13,460	4,096	89.6MB	2.38%	metricbeat-7.7.0-2020.06.29-000002

Hardening

Hardening Against Vulnerable Ports 22 and 80 on Target 1

Close port 22 and use port 443 with https instead of 80.

- Port 22 will prevent open ssh access to the machine. Using port 443 will provide a layer of security using ssl instead of the open port.
- Port 80 and 22 can be shut down with:
  - sudo ufw deny PORT 80
  - sudo ufw deny PORT 22
  - Sudo ufw allow PORT 443
    - Each command should be run one at a time and checked status with
      - sudo ufw status verbose

Hardening Against Weak/Insecure Passwords on Target 1

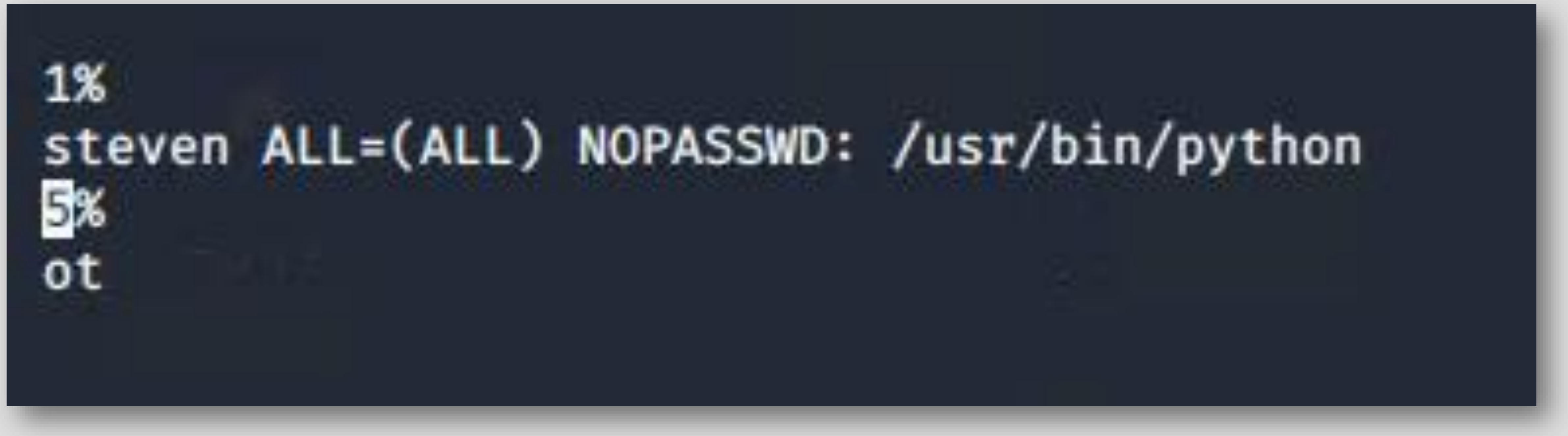
Users should change passwords to a best practices format involving at least 16 characters, no dictionary words, special characters, numbers and symbols. 1 hour lock outs should be implemented after 5 unsuccessful attempts within 15 minutes. Multi-factor authentication should also be used.

- Complex passwords are difficult to crack with brute force and lockouts will prevent multiple attempts. Additionally, notification alerts could be generated to further protect the accounts
- Install following the processes and recommendations at: https://ostechnix.com/how-to-set-password-policies-in-linux/

Hardening Against Python Privilege Escalation on Target 1

Python privileges should be removed for users vulnerable to ssh as well as users who are not authorized for root privileges.

- Removing the python sudo privileges will eliminate the potential for circumventing access restrictions
- vi /etc/sudoers
    - Delete this line: steven ALL=(ALL) NOPASSWD: /usr/bin/python



```
1%
steven ALL=(ALL) NOPASSWD: /usr/bin/python
5%
ot
```

Hardening Against Enumerate Wordpress Site on Target 1

Deploy the Ansible Playbook that updates the Wordpress site to a patched version with Stop User Enumeration plug-in and adjust firewall to block similar behaviors of enumerating traffic

- Updated versions Wordpress won't allow enumeration with appropriate plugins
- Run the ansible playbook discussed in the concluding slide and make sure Stop User Enumeration plug-in is installed and enabled
- https://wordpress.org/plugins/stop-user-enumeration/
- `sudo ansible-playbook -v WPandApache.yml`

Hardening Against Apache 2.4.10 CVE-2016-4975 on Target 1

Regularly update Apache server to latest stable version

- Apache tends to have significant vulnerabilities with every version. To keep ahead of these threats, it is important to maintain a consistent approach to upgrading the versions
- Run the ansible playbook discussed in the concluding slide

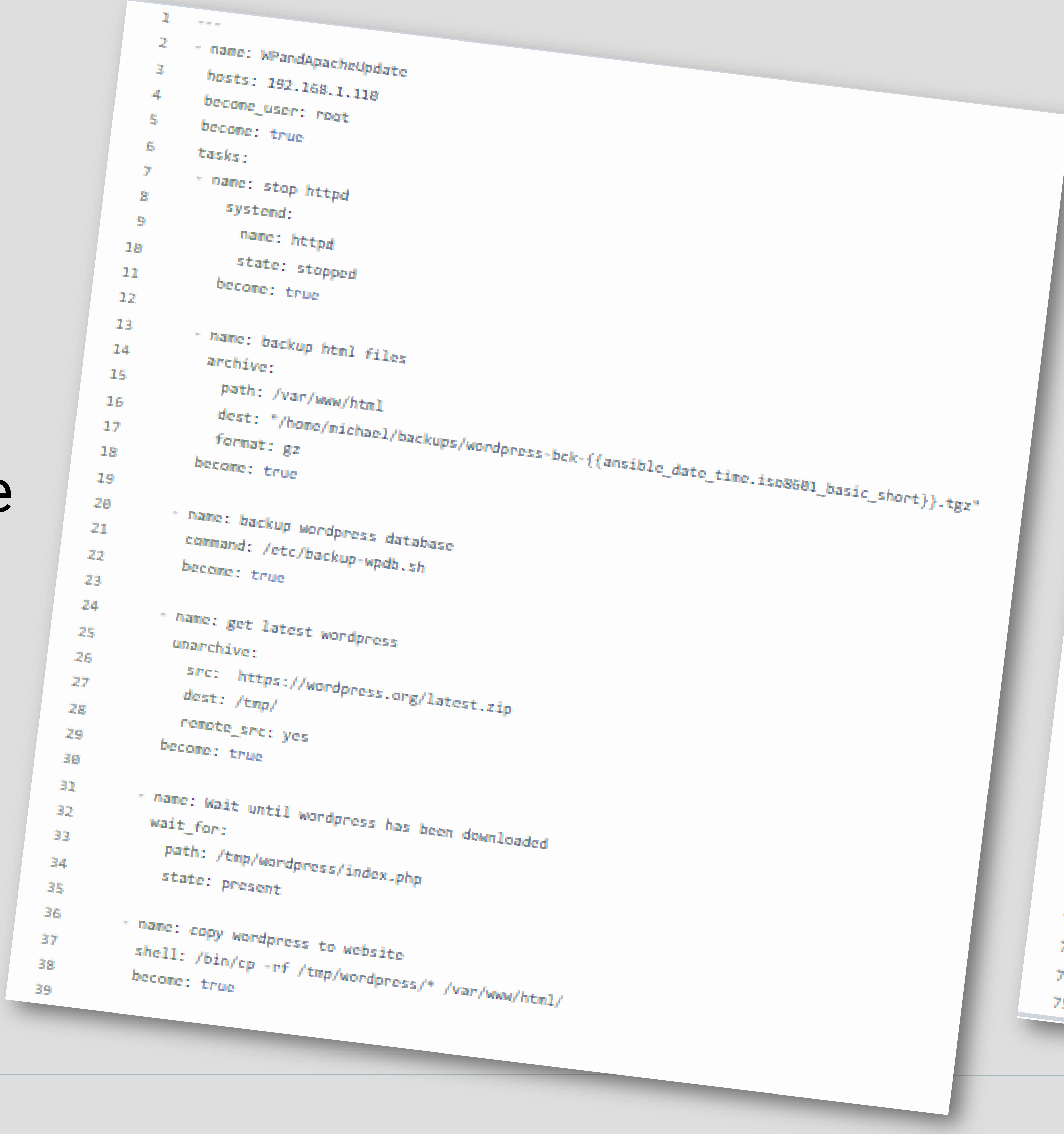
Implementing Patches

Implementing Patches with Ansible

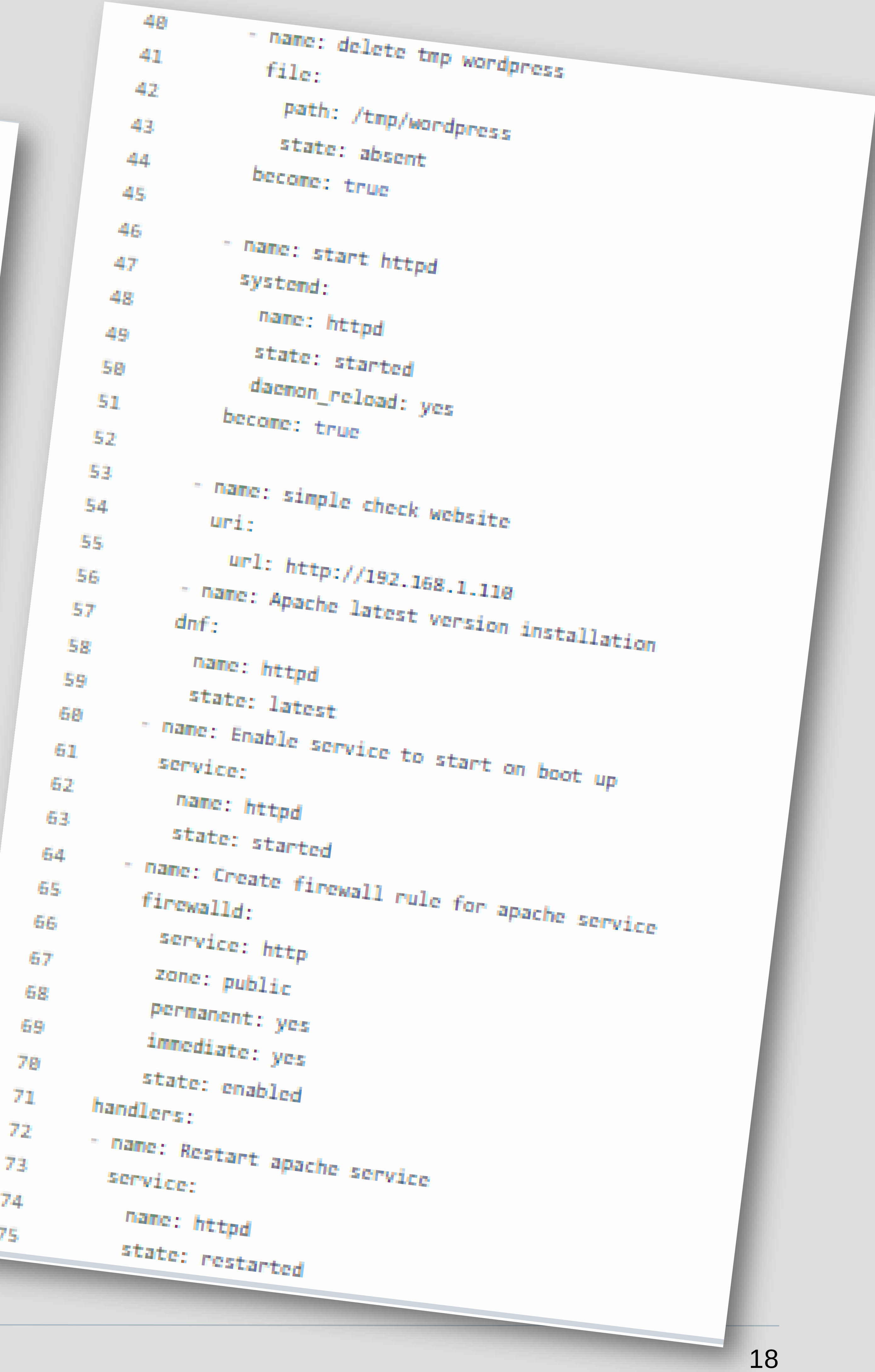
Playbook Overview

- Lines 7-55 update the wordpress html files and check the website

- Lines 56-75 update the Apache Server



```
1  ---
2  - name: WPandApacheUpdate
3    hosts: 192.168.1.110
4    become_user: root
5    become: true
6    tasks:
7    - name: stop httpd
8      systemd:
9        name: httpd
10       state: stopped
11     become: true
12
13   - name: backup html files
14     archive:
15       path: /var/www/html
16       dest: "/home/michael/backups/wordpress-bck-{{ansible_date_time.iso8601_basic_short}}.tgz"
17       format: gz
18     become: true
19
20   - name: backup wordpress database
21     command: /etc/backup-wpdb.sh
22     become: true
23
24   - name: get latest wordpress
25     unarchive:
26       src:  https://wordpress.org/latest.zip
27       dest: /tmp/
28       remote_src: yes
29     become: true
30
31   - name: Wait until wordpress has been downloaded
32     wait_for:
33       path: /tmp/wordpress/index.php
34       state: present
35
36   - name: copy wordpress to website
37     shell: /bin/cp -rf /tmp/wordpress/* /var/www/html/
38     become: true
39
```



```
40   - name: delete tmp wordpress
41     file:
42       path: /tmp/wordpress
43       state: absent
44     become: true
45
46   - name: start httpd
47     systemd:
48       name: httpd
49       state: started
50       daemon_reload: yes
51     become: true
52
53   - name: simple check website
54     uri:
55       url: http://192.168.1.110
56   - name: Apache latest version installation
57     dnf:
58       name: httpd
59       state: latest
60   - name: Enable service to start on boot up
61     service:
62       name: httpd
63       state: started
64   - name: Create firewall rule for apache service
65     firewalld:
66       service: http
67       zone: public
68       permanent: yes
69       immediate: yes
70       state: enabled
71   handlers:
72   - name: Restart apache service
73     service:
74       name: httpd
75       state: restarted
```