Blue Team: Summary of Operations

- Background
  - Network Topology
    - Kali
    - Target 1
    - Target 2
  - Description of Targets
  - Monitoring the Targets
    - Alert 1
    - Alert 2
    - Alert 3
  - Pattern of Behavior
  - Suggestions
    - Vulnerability 1
    - Vulnerability 2
    - Vulnerability 3

Background

We are working as a Security Engineer for X-CORP, supporting the SOC infrastructure. The SOC analysts have noticed some discrepancies with alerting in the Kibana system and the manager has asked the Security Engineering team to investigate.

To start, my team needs to confirm that newly created alerts are working. Once the alerts are verified to be working, we will monitor live traffic on the wire to detect any abnormalities that aren't reflected in the alerting system.

We will then report back all your findings to both the SOC manager and the Engineering Manager with appropriate analysis.

Network Topology

The following machines were identified on the network:

- **Kali**
  - Operating System: Kali Linux
  - Purpose: Attack
  - IP Address: 192.168.1.90

- **Target 1**
  - Operating System: Debian / GNU Linux
  - Purpose: Target
  - IP Address: 192.168.1.110

- **Target 2**
  - Operating System: Debian / GNU Linux
  - Purpose: Target
  - IP address: 192.168.1.115

Description of Targets

The target of this attack was: `Target 1` (`192.168.1.110`).

- `Target 1` is an Apache web server and has `SSH` enabled, so ports `80` and `22` are possible ports of entry for attackers. As such, the following alerts have been implemented.

Monitoring the Targets

Traffic to these services should be carefully monitored. To this end, we have implemented the alerts below.

Excessive HTTP Errors:

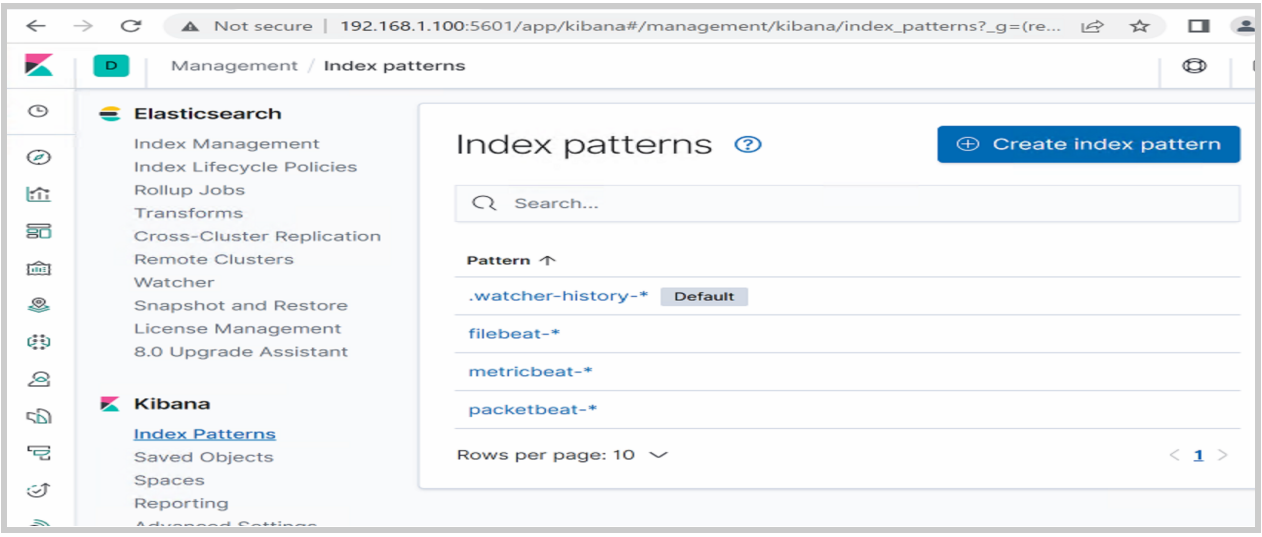
- **Alert 1**
  - Metric: Excessive HTTP Errors
  - Threshold: `WHEN count() GROUPED OVER top 5 'http.response.status_code' IS ABOVE 400 FOR THE LAST 5 minutes`
  - Vulnerability Mitigated: Bruteforce Attack
  - Reliability: High reliability, as 400 means there is an error. There will not be a lot of false positives or false negatives.

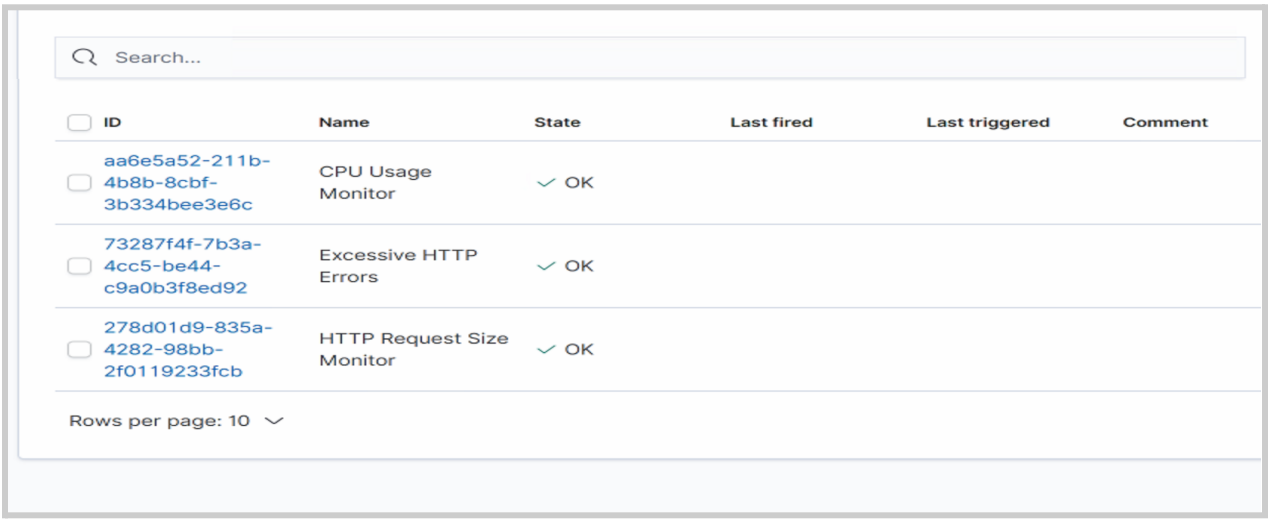
HTTP Request Size Monitor:

- **Alert 2**
  - Metric: HTTP Request Size Monitor
  - Threshold: `WHEN sum() of http.request.bytes OVER all documents IS ABOVE 3500 FOR THE LAST 1 minute`
  - Vulnerability Mitigated: Denial Service and DNS scanning
  - Reliability: Medium reliability because packets of a large size aren't necessarily malicious. A few false positives.

CPU Usage Monitor:

- **Alert 3**
  - Metric: CPU Usage Monitor
  - Threshold: `WHEN max() OF system.process.cpu.total.pct OVER all documents IS ABOVE 0.5 FOR THE LAST 5 minutes`
  - Vulnerability Mitigated: Denial of service
  - Reliability: Low reliability because a legitimate application can have high CPU usage. A lot of false positives may occur.


Not secure | 192.168.1.100:5601/app/kibana#/management/kibana/index_patterns?_g=(re...
Management / Index patterns
Elasticsearch
Index Management
Index Lifecycle Policies
Rollup Jobs
Transforms
Cross-Cluster Replication
Remote Clusters
Watcher
Snapshot and Restore
License Management
8.0 Upgrade Assistant
Kibana
Index Patterns
Saved Objects
Spaces
Reporting
Index patterns
Create index pattern
Search...
Pattern
.watcher-history-* Default
filebeat-*
metricbeat-*
packetbeat-*
Rows per page: 10
1


Search...
ID
Name
State
Last fired
Last triggered
Comment
aa6e5a52-211b-4b8b-8cbf-3b334bee3e6c
CPU Usage Monitor
OK
73287f4f-7b3a-4cc5-be44-c9a0b3f8ed92
Excessive HTTP Errors
OK
278d01d9-835a-4282-98bb-2f0119233fcb
HTTP Request Size Monitor
OK
Rows per page: 10


Discover
New
Save
Open
Share
Inspect
result.condition.met :true
KQL
~ 7 days ago → now
+ Add filter
.watcher-history-*
Search field names
Filter by type
0
Selected fields
_source
Available fields
_id
_index
_score
_type
condition.script.la...
147 hits
May 27, 2022 @ 19:33:08.813 - Jun 3, 2022 @ 19:33:08.814 —
Auto
Count
40
30
20
10
0
2022-05-28 00:00
2022-05-29 00:00
2022-05-30 00:00
2022-05-31 00:00
2022-06-01 00:00
2022-06-02 00:00
2022-06-03 00:00
trigger_event.triggered_time per 3 hours
Time
_source
Jun 3, 2022 @ 19:29:16.767
watch_id: aa6e5a52-211b-4b8b-8cbf-3b334bee3e6c node: 0dGY9fonT-ysLBrBCvSF-g state: execute
status.state.active: true status.state.timestamp: 2022-06-03T19:10:29.327Z status.last_che

Suggestions

- Each alert above pertains to a specific vulnerability/exploit. Recall that alerts only detect malicious behavior, but do not stop it.
    - For each vulnerability/exploit, identified by the alerts mentioned above, suggest a patch (e.g. implementing a blocklist as an effective tactic against brute-force attacks).

- Logs and alerts generated during our assessment suggest the network is susceptible and must be hardened against several active threats, identified by the alerts mentioned above.

- The Blue Team suggests staff IT implement the fixes below to protect the network:

Vulnerability 1

- Patch: Block repetitive login failure by blocking IPs and adding block out policies
- Why It Works: Because we block potentially malicious traffic and minimize brute force attack potential

Vulnerability 2

- Patch: Block traffic large packet size
- Why It Works: Because we block potentially malicious traffic

Vulnerability 3

- Patch: Monitor CPU usage and send alert when threshold is exceeded
- Why It Works: Because the usage can be tracked
