ДОДАТОК А

Конфігураційний файл, який використовується Vagrant для розгортання необхідну кількість машин з заданими параметрами

```
#  Ansible
Vagrant.configure("2") do |config|
  config.vm.define "ansible" do |ansible|
    ansible.vm.box = "ubuntu/bionic64"
    ansible.vm.box_check_update = false
    ansible.vm.hostname = "ansible"
    ansible.vm.network "public_network", type: "dhcp", bridge: "Realtek PCIe GBE Family
Controller"
          ansible.vm.provider :virtualbox do |boost|
      boost.customize ['modifyvm', :id, '--memory', 2048]
      boost.customize ["modifyvm", :id, "--cpus", "2"]
    end
  end

 # Elasticserch and Kibana
  config.vm.define "ek" do |ek|
    ek.vm.box = "ubuntu/bionic64"
    ek.vm.box_check_update = false
    ek.vm.hostname = "ek"
    ek.vm.network "public_network", type: "dhcp", bridge: "Realtek PCIe GBE Family
Controller"
    ek.vm.provider :virtualbox do |ad|
      ad.customize ['modifyvm', :id, '--memory', 4096]
      ad.customize ["modifyvm", :id, "--cpus", "2"]
    end
  end
```

```
 # Logstash
  config.vm.define "logstash" do |logstash|
    logstash.vm.box = "ubuntu/bionic64"
    logstash.vm.box_check_update = false
    logstash.vm.hostname = "logstash"
    logstash.vm.network "public_network", type: "dhcp", bridge: "Realtek PCIe GBE Family Controller"
    logstash.vm.provider :virtualbox do |ad|
      ad.customize ['modifyvm', :id, '--memory', 2048]
      ad.customize ["modifyvm", :id, "--cpus", "2"]
    end
  end
   # Server 1
  config.vm.define "server1" do |server1|
    server1.vm.box = "ubuntu/bionic64"
    server1.vm.box_check_update = false
    server1.vm.hostname = "server1"
    server1.vm.network "public_network", type: "dhcp", bridge: "Realtek PCIe GBE Family Controller"
  end
end
```

ДОДАТОК Б

Ansible playbook який використовується для розгортання ELK

```
---
- name: Install Elasticsearch
  hosts: ek
  become: True
  roles:
    - java
    - elasticsearch
    - kibana
- name: Install Filebeat
  hosts:
    server1
  become: True
  roles:
    - filebeat
- name: Install Logstash
  hosts: logstash
  become: True
  roles:
    - java
    - logstash
```

ДОДАТОК В

Ansible inventory файл

```
ek ansible_host=192.168.43.186 ansible_user=vagrant
ansible_ssh_private_key_file=/home/vagrant/.ssh/id_rsa
logstash ansible_host=192.168.43.88 ansible_user=vagrant
ansible_ssh_private_key_file=/home/vagrant/.ssh/id_rsa
server1 ansible_host=192.168.43.67 ansible_user=vagrant
ansible_ssh_private_key_file=/home/vagrant/.ssh/id_rsa
```

ДОДАТОК Г

Ansible роль, для встановлення Elasticsearch

```
---
- name: Update
  apt:
    name: aptitude
    update_cache: yes
- name: import the Elasticsearch public GPG key into APT
  apt_key:
    url: https://artifacts.elastic.co/GPG-KEY-elasticsearch
    state: present
- name: add the Elastic source list to the sources.list.d
  shell: echo "deb https://artifacts.elastic.co/packages/6.x/apt stable main" | sudo tee -a /etc/apt/sources.list.d/elastic-6.x.list
- name: Update
  apt:
    name: aptitude
    update_cache: yes
- name: Install elasticsearch
  apt:
    name: elasticsearch
    state: latest
- name: Copy file with configuration
  copy:
    src: "{{ config_file }}"
    dest: "{{ elasticsearch_folder }}"
    owner: root
    group: elasticsearch
    mode: 0660
```

```
- name: Starting and enable elasticsearch
  service:
    name: elasticsearch
    state: started
    enabled: yes
```

ДОДАТОК Д

Файл конфігурації Elasticsearch

```
# ---------------------------------- Paths ------------------------------------
# Path to directory where to store the data (separate multiple locations by comma):
#
path.data: /var/lib/elasticsearch
#
# Path to log files:
#
path.logs: /var/log/elasticsearch
#
# ---------------------------------- Memory -----------------------------------
# Lock the memory on startup:
#
bootstrap.memory_lock: true
#
# ---------------------------------- Network ----------------------------------
# Set the bind address to a specific IP (IPv4 or IPv6):
#
network.host: 0.0.0.0
#
# Set a custom port for HTTP:
#
http.port: 9200
#
# ---------------------------------- Gateway ----------------------------------
# Block initial recovery after a full cluster restart until N nodes are started:
#
gateway.recover_after_nodes: 3
```

ДОДАТОК Е

Ansible роль, для встановлення Kibana

```
---
- name: Update
  apt:
    name: aptitude
    update_cache: yes
- name: Install Nginx
  apt:
    name: nginx
    state: latest
- name: Install Kibana
  apt:
    name: kibana
    state: latest
- name: Updating the config file to allow outside access
  lineinfile:
   destfile: /etc/kibana/kibana.yml
   regexp: 'server.host:'
   line: 'server.host: 0.0.0.0'
- name: Defining server port
  lineinfile:
   destfile: /etc/kibana/kibana.yml
   regexp: 'server.port:'
   line: 'server.port: 5601'
- name: Defining Elasticsearch URL
  lineinfile:
   destfile: /etc/kibana/kibana.yml
   regexp: 'elasticsearch.hosts:'
```

```
    line: 'elasticsearch.url: "http://{{ elasticsearch_ip }}:9200"'
- name: Starting and enable kibana
  service:
    name: kibana
    state: started
    enabled: yes
- name: Copy file with configuration
  copy:
    src: "{{ config_file_nginx }}"
    dest: "{{ destin_folder_nginx }}"
    owner: root
    group: root
    mode: 0644
- name: Create a symbolic link
  file:
    src: /etc/nginx/sites-available/kibana
    dest: /etc/nginx/sites-enabled/default
    owner: root
    group: root
    state: link
- name: Starting and enable nginx
  service:
    name: nginx
    state: started
    enabled: yes
- service:
    name: nginx
    state: restarted
```

ДОДАТОК Ж

Файл конфігурації для Kibana

```
server {
    listen 80;

    server_name kibana.com;

    location / {
        proxy_pass http://localhost:5601;
        proxy_http_version 1.1;
        proxy_set_header Upgrade $http_upgrade;
        proxy_set_header Connection 'upgrade';
        proxy_set_header Host $host;
        proxy_cache_bypass $http_upgrade;
    }
}
```

ДОДАТОК И

Ansible роль, для встановлення Logstash

```
---
- name: Update
  apt:
    name: aptitude
    update_cache: yes
- name: Add the GPG key to install signed packages
  apt_key:
    url: https://artifacts.elastic.co/GPG-KEY-elasticsearch
    state: present
- name: Install the apt-transport-https package
  apt:
    name: apt-transport-https
- name: Add the Elastic package repository to repository list
  shell: echo "deb https://artifacts.elastic.co/packages/7.x/apt stable main" | sudo tee -a /etc/apt/sources.list.d/elastic-7.x.list
- name: Updatee
  apt:
    name: aptitude
    update_cache: yes
- name: Install Logstash
  apt:
    name: logstash
- name: Copy file with configuration
  copy:
    src: "{{ item }}"
    dest: "{{ destin_folder }}"
    owner: root
```

```
    group: root
    mode: 0644
  with_fileglob:
    - "02-beats-input.conf"
    - "10-syslog-filter.conf"
- name: Copy template file
  template:
    src: 30-elasticsearch-output.j2
    dest: "{{ destin_folder }}/30-elasticsearch-output.conf"
- name: Enable logstash
  service:
    name: logstash
    state: started
    enabled: yes
```

ДОДАТОК К

Файл конфігурації для Logstash

```
filter {
  if [type] == "syslog" {
    grok {
      match => { "message" => "%{SYSLOGTIMESTAMP:syslog_timestamp}
%{SYSLOGHOST:syslog_hostname}
%{DATA:syslog_program}(?:\[%{POSINT:syslog_pid}\])?:
%{GREEDYDATA:syslog_message}" }
      add_field => [ "received_at", "%{@timestamp}" ]
      add_field => [ "received_from", "%{host}" ]
    }
    syslog_pri { }
    date {
      match => [ "syslog_timestamp", "MMM  d HH:mm:ss", "MMM dd HH:mm:ss" ]
    }
  }
}
```