Данный проект предназначен для развёртывания VPN сервера и доступа клиентов в локальную и всемерную сеть.

В проекте используются следующие компоненты:

|  |  |
| --- | --- |
| ОС | Ubuntu Linux |
| Облачный провайдер | Google Cloud |
| Используемые пакеты приложений | OpenVPN  Easy-RSA  Expect  prometheus  prometheus-node-exporter  Bash  Rclone |

Система развертывания организована следующим образом:

**1.** При запуске скрипта bash инициируются переменные - в частности расположение конфигурации и сертификатов на системе администратора (Linux система).

**2.** Проверится наличие утилиты "gcloud", в случае отутствия устанавливается, предлагает инструкцию по настройке авторизации в Google Cloud и прекращает работу пока не будет выполнена настройка доступа к Google Cloud средствами "gcloud init"

**3.** В случае выполненной авторизации в Google Cloud проверяется наличие каталога для конфигурации и сертификатов на системе администратора и выполняется его очистка для копирования новой конфигурации

**4.** После успешного выполнения предыдущих проверок выполняется настройка файрволла на стороне Google Cloud для публичного доступа к Prometheus и OpenVPN

**5.** Далее средствами утилиты "gcloud" создаются виртуальные машины с последующей настройкой программного обеспечения

**5.1.** Создается PKI-Server, выполняется обновление ОС, установка необходимых пакетов и настройка центра сертификации, автоматически генерируется парольная фраза к паре открытого и закрытого ключа ЦС.

- Парольная фраза будет сохранена в файле ca.txt, с последующим копирование данного файла на систему администратора в каталог конфигураций

- Далее будет сконфигурированы сервисы для резервного копирования - backup.service и backup.timer. Их задача выполнить резервное копирование в указанное время локально и перенести РК в облачное хранилище

- Облачное хранилище так же настраивается автоматически и использует ПО rclone

**5.2.** Следующим этапом создается и настраивается OpenVPN сервер.

- На OpenVPN-Server выполняется обновление ОС, установка необходимых пакетов и настройка ПО

- Генерируются запросы на сертификат для OpenVPN сервера и переносятся на систему администратора в каталог конфигураций, далее данный запрос на сертификат переносится на PKI-Server

- Выполняется подпись запроса на сертификат OpenVPN сервера сервером ЦС и перенос подписанного сертификата на сервер OpenVPN

- Выполняется настройка OpenVPN сервера средствами скрипта vpn.sh - настройка файрволла, разрешение перенаправления пакетов и т.д. vpn.sh - будет скачан с git репозитария

- Далее будет сконфигурированы сервисы для резервного копирования - backup.service и backup.timer. Их задача выполнить резервное копирование в указанное время локально и перенести РК в облачное хранилище

- Облачное хранилище так же настраивается автоматически и использует ПО rclone.

- Выполняется перезагрузка сервера

**5.3.** Следующим этапом создается и настраивается Prometheus-Server сервер

- На Prometheus-Server выполняется обновление ОС, установка необходимых пакетов и настройка ПО

- Выполняется настройка Prometheus сервера /etc/prometheus/prometheus.yml

- Выполняется настройка Prometheus Alert Manager-а /etc/prometheus/alertmanager.yml, /etc/prometheus/alert.rules.yml

- Выполняется перезапуск сервисов для применения конфигурации Prometheus и Prometheus Alert Manager-а

**6.** Выполняется поиск локальных ip адресов виртуальных машин и вносятся в hosts файлы

**7.** Генерация клиентской конфигурации OpenVPN

- Для генерации конфигурации клиента используется скрипт vpnclient.sh

- vpnclient.sh создает временный скрипт tempclient.sh и подменяет имя клиента, указанного в аргументе для скрипта vpnclient.sh

- выполняется подключение к серверу OpenVPN

- далее скрипт генерирует файл запроса на сертификат

- файл запроса переносится на PKI-Server и подписывается

- сертификата и ключ клиента переносятся на OpenVPN сервер

- далее make\_config.sh генерирует конфигурацию OpenVPN клиента в виде файла - client.ovpn с внесением всех ключей и сертификатов

- далее файл переносится в каталог конфигураций на системе администратора

Связи компонентов

