Заметки к лаб работам

В корневом каталоге имеется каталог **`/ap-2301`**, в котором находятся файлы, которые вам могут потребоваться в ходе выполнения заданий.

Модуль 2.

Здесь и далее в виртуальных машинах в source.list указанны локальные репозитории, которые синхронизируются с репозиториями Astra Linux. Не меняйте их. Это сделано для ускорения процедуры установки пакетов.

Для всех контроллеров домена устанавливайте 8Гб ОЗУ и 4 ядра. Команда для развертывания КД должна выглядеть так:

```
sudo aldpro-server-install -d ald.test -n dc01 --ip 192.168.101.201 \
-p P@ssw0rd --no-reboot --setup_syncer --setup_gc
```

Перед первым входом в веб-интерфейс ALDPro откройте файл `/usr/lib/firefox/distribution/policies.json` и поменяйте в нем URL с `https://dc01/` на `https://dc01.ald.test/` - это сделает работу с веб-порталом удобней.

По окончанию лабораторной работы добавьте добавьте в домен еще 2 узла:

- monitor: 4Гб ОЗУ, 2 ядра;
- log: 2Гб ОЗУ, 2 ядра.

Эти узлы нам потребуются для выполнения некоторых заданий. Потом в модуле 8 мы на них установим соответствующие роли. Вы можете воспользоваться скриптом `/ap-2301/client_install.sh` для автоматизации добавления новых машин в домен.

Модуль 3.

Предварительно требуется создать пользователей командой:

```
ipa user-add --first=Ivan --last=Ivanov --password ivanov
ipa user-add --first=User --last=First --password user
```

После выполнения лабораторной работы удалите пользователей:

```
ipa user-del ivanov
ipa user-del user
```

Модуль 4.

Лабораторную работу следует выполнять после работ модулей 5, 6 и 7, в которых изучается соответственно создание пользователей, политик и заданий автоматизации.

Узлов dc03, dhcp и client2 пока нет, но появятся позже. Картинка со схемой и таблица под ней не правильные, содержат лишнюю информацию.

Модуль 6.

Обратите внимание на пункты 3 и 4 задания по политикам повышения привилегий. Не забудьте предоставить право HBAC для команды sudo.

По завершению задания «Создание политики повышения привилегий» включите правило HBAC **allow_all**.

Перед началом задания создайте веб-хранилище для файлов salt. Инструкция ниже. Во всех скриптах политик и автоматизации используйте в качестве источника: - `source: https://saltfs/` + `skip_verify: true` вместо `salt:///files`.

Для принудительного обновления политик вы можете использовать скрипт `gpupdate`. Его следует запускать через `sudo`. Скрипт нужен только для версии 2.3. В версиях 2.4+ имеется команда `aldpro-gpupdate`. Содержимое скрипта:

```
#!/bin/bash

# Check first argument

if ! echo "$1" | egrep -q '^(gp|swp|audit)$'

then

 echo -e '\e[31;5mError\e[0m'

 echo -e "Usage:\n$0 { gp | swp | audit }"

 exit 1

fi


# Update cache

salt-call state.apply gpupdate.build -c /srv/salt/standalone/config
pillar='{"verbose": True, "force": True}'


# Run GP Update

salt-call state.apply gpupdate.$1 -c /srv/salt/standalone/config/
pillar='{"force": True,"verbose": True}'
```

Настройка веб хранилища для salt файлов

Подготовка веб сервера

1. Создайте новую ВМ (клон). Память 2Г., 2 ядра.
2. Добавьте эту машину в домен с именем saltfs.
3. Установите на машину пакет apache2.
4. Выключите режим Astramode в веб-сервере; надо в файле `/etc/apache2/apache2.conf` добавить строку:

```
AstraMode off
```

5. На контроллере домена создать сертификат для веб сервера командой:

```
sudo astra-freeipa-server-crt --host saltfs.ald.test
```

6. Скопировать сертификат на машину saltfs в файл `/etc/ssl/certs/saltfs.cert`.
7. Скопировать ключ на машину saltfs в файл `/etc/ssl/private/saltfs.key`.
8. В файле `/etc/apache2/sites-available/default-ssl.conf` следующие строки привести к виду:

```
SSLCertificateFile /etc/ssl/certs/saltfs.cert
SSLCertificateKeyFile /etc/ssl/private/saltfs.key
SSLCACertificateFile /etc/ipa/ca.crt
```

9. Включить модуль ssl:

```
a2enmod ssl
```

10. Включить сайт default-ssl:

```
a2ensite default-ssl.conf
```

11. Перезапустить веб сервер.

```
sudo systemctl restart apache2
```

12. Удалить файл `/var/www/html/index.html` (можно его не удалять, но тогда при входе на сервер мы не увидим файлы).
13. Создать каталоги, в которых будут размещаться файлы указанные в политиках и заданиях автоматизации, использующиеся в политиках. Например:

```
mkdir /var/www/html/gpo
mkdir /var/www/html/automations
```

Настройка политик и заданий автоматизации для передачи файлов

см. https://docs.saltproject.io/en/latest/ref/states/all/salt.states.file.html#salt.states.file.managed

Вариант 1

1. Использовать вложенное описание содержимого файла. Например:

```
/path/to/file1:
  file.managed:
    - contents:
      - This is line 1
      - This is line 2


/path/to/file2:
  file.managed:
    - contents: |
        This is line 1
        This is line 2
```

2. Не может быть использовано в комбинации с `source`.
3. Не использует движка шаблонов.

Вариант 2

1. Использовать внешнюю ссылку на http(s) сервер. Например:

```
test_file:
  file.managed:
    - name: /tmp/test.txt
    - source: https://saltfs/dir1/test.txt
    - source_hash: 79eef25f9b0b2c642c62b7f737d4f53f
```

2. Требуется обязательно указывать или опцию `source_hash:`, или `skip_verify: true`.