

Automate The Creation of VAPs With The Realm Python Library

Goal: Create a python script to create VAPs

Using the `realm.py` library we will write a script that will allow us to automate the creation of VAPs.
Requires LANforge 5.4.2

1. Building a VAP

A. VAPProfile

A. The preferred method for creating a `vap_profile` is to use the factory method `new_vap_profile()` found in `realm`

I. We will need to set the name of our vap using `vap_profile.vap_name`

Example:

```
vap_profile.vap_name = "TestNet"
```

II. `vap_profile.use_security(security_type, ssid, passwd)` is the preferred method to use when setting the security type, ssid, and password variables.

Available security types are wpa, wpa2, wpa3, wep, and open.

Example:

```
vap_profile.use_security(type="wpa2", ssid="testNet", passwd="testPass")
```

III. We can change the mode at any time before calling `create()` by modifying the `vap_profile.mode` variable. Changing the mode will allow us to specify the 802.11 wireless standard the VAP uses. See [here](#) for available modes.

Example:

```
vap_profile.mode = 1
```

IV. The channel to be used by the VAP can be set with the channel parameter of the `create()` method.

Example:

```
vap_profile.create(resource=1, radio="wiphy0", channel=36, up_=True)
```

2. Bringing VAPs Up/Down

A. `vap_profile.admin_up()` and `vap_profile.admin_down()` can be used to bring the VAP up or down, as necessary.

3. Using TTLS

A. TTLS setup requires a few pieces of information to work correctly. `VAPProfile` has a `set_wifi_extra()` method for setting the relevant variables. See [here](#) for the available options

- B. We will need a key management type (`key_mgmt`), an EAP method (`eap`), an EAP identity string (`identity`), an EAP password string (`passwd`), an 802.11u realm (`realm`), an 802.11u domain (`domain`), and an 802.11u HESSID (`hessid`)

Example:

```
key_mgmt="WPA-EAP"
eap="TTL"
identity="testuser"
passwd="testpasswd"
realm="localhost.localdomain"
domain="localhost.localdomain"
hessid="00:00:00:00:00:01"
```

We can then use these variables to call the `set_wifi_extra()` method

Example:

```
vap_profile.set_wifi_extra(key_mgmt, eap, identity, passwd, realm, domain, hessid)
```

4.

Cleaning Up

- A. `vap_profile.cleanup()` can be used to remove any VAPs that were created by the profile