Openvpn User Acess Manager

OUAM Project

Specifications :

This tool must manage several notions about the authentication of the users of a VPN service. The service in question will be a OpenVPN instance, it's an open source implementation of a VPN protocole that allows two distant points to be connected by a secure network link.

The following list, enumerates all specific points on which the project will repose :

- Connection : Each users **must have its own credentials**. A credential is considered as an authentication method. This can take two form, first it can be represented by an uniq **certificat** or by some **login/password**.
- Authentication : Credentials must have a reasonable **lifetime** (customizable) in order to limit the potentials security breach if the credentials are losted. In the same think, this tool must implement a way to **make new credential** automatically and **provides them to differents** users in a minimum time to maintain the continuity of service.
- Profil : Each user must be represented by a **profil** which contains all data allowing their identification as theirs **settings**.
- Security :
  - the system must be master of all VPN usager, this means that each **connection must be checked** according to current situation
  - it must keep a trace of all **connection** by a systematic logging
  - it must keep also some **traces of all executed operations** (credential generation, expiry of credentials, credentials sending to user)
  - intercept all **intrusion** attempt and make **alert to an administrator**
- Assistance : the system must contains a helping module, that provide some tips to user when some specific errors occurs. This module can take the following form :
  - a manual

    - a **configuration error detector** that can send some notify to the user with some advice to resolve its problem
- API : The apps must implement an entry point, as type of API REST, that allow other applications to make queries on this about system status.

Constraints

The system must be as modular as possible because it may have the capability to be connected with some others applications in a greater Information System.

The data storage system will be in a SQL format in first time.

The program will be realized as a single executable and must be written into a advanced language facilitating shell command execution.

The program must be running into a limited environnement of type CHROOT.

First release :

1. The base engine of the program (base daemon program + modular features)
2. Users profiles definition and profil management classes
3. Credential generator engine
4. Link between profils ⇔ credentials generation