**Tự động khởi động chương trình Python trên Linux https://deha.ai/blog/tu-dong-khoi-dong-chuong-trinh-python-tren-linux/**

**24 Th5 2019 By: Tuyen**

Trong bài viết này, tôi sẽ hướng dẫn bạn cách chạy một chương trình Python sử dụng Systemd trong Linux. Khi sử dụng Systemd, bạn có thể bật/tắt/khởi động chương trình Python của mình như một dịch vụ của hệ thống.

**Bước 1: Chuẩn bị một file python làm ví dụ.**



|  |  |
| --- | --- |
| 1 | sudo vi /usr/bin/dummy\_service.py |

Sau đó thêm nội dung sau vào File. Bạn hoàn toàn có thể tự viết mã Python mà bạn muốn.



|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12 | #!/usr/bin/python3  import socket  s = socket.socket(socket.AF\_INET, socket.SOCK\_STREAM)  s.bind(("localhost", 9988))  s.listen(1)  while True:      conn, addr = s.accept()      data = conn.recv(1024)      conn.close()      my\_function\_that\_handles\_data(data) |

**Bước 2: Tạo Service file**

Tạo một file server nằm trong thư mục /lib/systemd/system của hệ thống. Lưu ý file này phải có phần mở rộng là .service



|  |  |
| --- | --- |
| 1 | sudo vi /lib/systemd/system/dummy.service |

Sau đó thêm nội dung vào file này như sau



|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12 | [Unit]  Description=Dummy Service  After=multi-user.target  Conflicts=getty@tty1.service    [Service]  Type=simple  ExecStart=/usr/bin/python3 /usr/bin/dummy\_service.py  StandardInput=tty-force    [Install]  WantedBy=multi-user.target |

**Bước 3: Khởi tạo Service lần đầu**

Mỗi khi thêm 1 Service mới, bạn bắt buộc phải khởi động lại cơ chế deamon của Linux.



|  |  |
| --- | --- |
| 1 | sudo systemctl daemon-reload |

Tiếp theo, hãy bật Service mà mình vừa thêm để nó khởi động cùng với hệ thống. Tức là khi bạn bật/khởi động lại Server thì Service của chúng ta sẽ tự khởi động theo.



|  |  |
| --- | --- |
| 1  2 | sudo systemctl enable dummy.service  sudo systemctl start dummy.service |

Cuối cùng của bước này, hãy khởi động Service của bạn lên để nó hoạt động.



|  |  |
| --- | --- |
| 1 | sudo systemctl start dummy.service |

**Bước 4: Khởi động/tắt/trạng thái của Service.**

Để xem trạng thái của Service hiện tại, hãy dùng lệnh sau



|  |  |
| --- | --- |
| 1 | sudo systemctl status dummy.service |

Để khởi động/bật/tắt bằng tay Service của mình, hãy dùng các lệnh sau



|  |  |
| --- | --- |
| 1  2  3 | sudo systemctl stop dummy.service          #To stop running service  sudo systemctl start dummy.service         #To start running service  sudo systemctl restart dummy.service       #To restart running service |

**Kết**

Khá đơn giản nhưng hữu ích phải không nào? Ngoài Python bạn có thể sử dụng Systemd để chạy bất cứ chương trình nào như Docker, Kafka… hay các Script được viết bằng Ruby, Java, C++…