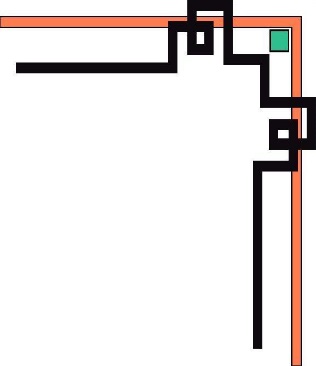
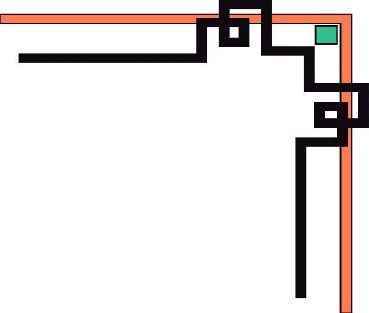
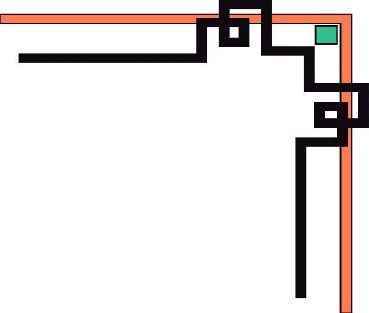
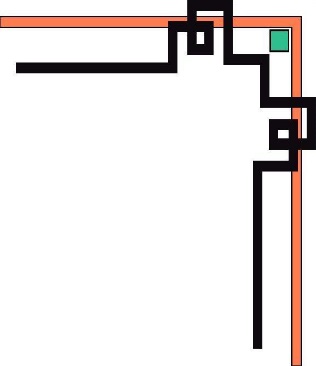
*ĐẠI HỌC DUY TÂN*



*KHOA ĐIỆN – ĐIỆN TỬ*



BÀI BÁO CÁO

**MÔN HỌC: Hệ điều hành Linux - Unix**

**ĐỀ TÀI: Tìm hiểu về Function**

*GVHD: Đặng Ngọc Cường*

*Lớp: CS 226 D*

*SVTH: Lê Duy Vương – 23211210077*

*Đỗ Hoàng Duy – 24211200229*

*Trần Văn Bình – 24211216815*

Đà Nẵng, tháng 4 năm 2020

**Function là gì?**

Func là một Method gói gọn các câu lệnh trong nó, để thực hiện một chức năng cụ thể.

Trong đó:

• Biểu tượng cho function chính là 2 dấu { } .

• Function được xác định thông qua function name

• Có giá trị trả về

• Không có hoặc có một hoặc nhiều tham số

Quan trọng

*Để thực thi một function thì phải cần tới toán tử ( ). Các giá trị đối số sẽ được gởi vào trong 2 dấu () đó*

Cú pháp

<từ khoá functuon> <function name> ( [danh sách tham số] ) -> <kiểu giá trị trả về> {

//code

return <giá trị>

}

\*

Đó là cú pháp đơn giản nhất để khai báo 1 function. Tuỳ thuộc vào từng ngôn ngữ lập trình mà sẽ có cách khai báo và từ khoá khác nhau. Ví dụ với Swift là func.

**Func:**

• Khai báo hàm (Function Declaration) (2)(3): gồm tên hàm, tham số (parameter) đầu vào (2), kiểu trả về (return type) (3)

• Phần thâm hàm (Function Definition) (4): nằm trong dấu {...}, chứa các câu lệnh ***for...in*** hoặc ***func***, ....để sử lý tham số đầu vào, **return** ra giá trị trả về

• Chỉ được sử dụng khi nó được gọi ra (1)

**Function được sử dụng rất nhiều trong code, và cách viết để sử dụng nó cũng rất đa dạng:**

• Không có tham số đầu vào nhưng vẫn có giá trị trả về

• Không có tham số đầu vào cũng không có tham số trả về

• Tham số đầu vào có thể có nhiều giá trị, một mảng (array), hoặc cả một func,... và giá trị trả về cũng vậy

• ...

**Một số lưu ý của func**

• Trong Function chỉ ***return*** một lần

• Một Function sau khi ***return*** xong sẽ được giải phóng bộ nhớ.

• Tham số truyền vào mặc định là hằng số - không đổi. Vì vậy để thay đổi được tham số (parameter) đầu vào trong func cần sử dụng ***inout*** và ***&***

Các dạng function phổ biến

1. Public & Private function

Đây chính là phạm vi truy cập tới cho function. Nếu bạn muốn các class khác không dùng được thì sử dụng private. Còn với public thì bên ngoài sẽ gọi được nó.

2. Tham số

Function còn được chia ra 2 dạng cơ bản nữa:

• Có tham số

• Không có tham số

Với có tham số thì sẽ được chia ra 2 loại cơ bản tiếp theo:

• Có một tham số

• Có nhiều tham số

3. Giá trị trả về

Cũng đươc chia ra thành 2 loại cơ bản:

• Giá trị trả về là void , tức là không có giá trị trả về

• Giá trị trả về khác void

Từ khoá được sử dụng chính là return. Khi chạy tới return thì function sẽ kết thúc.

4. Gọi hàm

Sự kết hợp giữa 2 yếu tố trên là tham số và giá trị trả về. Và cũng chia ra 2 dạng cơ bản:

• Gọi hàm đơn thuần

• Gọi hàm lồng nhau

Giá trị trả về của function này chính là giá trị cho tham số của function khác. Function có thể gọi trực tiếp từ việc truyền giá trị cho tham số của function khác.

**Func - Install Func**

**[1]**

**Cài đặt func cho Certmaster và Minion.**

# install from EPEL

**[**root@certmaster ~]# yum --enablerepo=epel -y install func

**[2]**

**Chạy certmaster trên Certmaster Server.**

**VD:**

[root@certmaster ~]# /etc/rc.d/init.d/certmaster start

Starting certmaster daemon: [ OK ]

[root@certmaster ~]# chkconfig certmaster on

**[3]**

**Đăng nhập vào chứng chỉ trên Certmaster Server.**

**VD:**

# list the host that requests certificates

[root@certmaster ~]# certmaster-ca --list

minion01

[root@certmaster ~]# certmaster-ca --sign minion01

/var/lib/certmaster/certmaster/csrs/minion01.csr signed - cert located at /var/lib/certmaster/certmaster/certs/minion01.cert

# make sure it works

[root@certmaster ~]# func "\*" list\_minions

minion01

**[4]**

**Chạy func trên Minion Server.**

[root@minion01 ~]# vi /etc/certmaster/minion.conf

# configuration for minions

[main]

# Certmaster's hostname or IP Address

certmaster = certmaster.srv.world

certmaster\_port = 51235

log\_level = DEBUG

cert\_dir = /etc/pki/certmaster

[root@minion01 ~]# /etc/rc.d/init.d/funcd start

Starting func daemon: [ OK ]

[root@minion01 ~]# chkconfig funcd on

**Func - Cách sử dụng?**

**Cách sử dụng lệnh cơ bản lệnh func.**

**[1] Show minons (hiển thị minons)**

# show all minions

[root@certmaster ~]# func "\*" list\_minions

minion01

minion02

# show specific minions

[root@certmaster ~]# func "minion01;minion02" list\_minions

minion01

minion02

**[2] Show all modules (hiển thị tất cả modules)**

# show modules on minion01

[root@certmaster ~]# func "minion01" call system list\_modules

{'minion01': ['bridge',

'certmastermod',

'command',

'confmgt\_augeas',

'copyfile',

'cpu',

'delegation',

'disk',

'djangoctl',

'echo',

'fact',

'filetracker',

'func\_getargs',

'func\_module',

'getfile',

'hardware',

'httpd',

'iptables',

'iptables.port',

'jboss',

'jobs',

'mount',

'nagios-check',

'netapp.options',

'netapp.snap',

'netapp.vol',

'netapp.vol.clone',

'networktest',

'overlord',

'process',

'pullfile',

'reboot',

'rpms',

'service',

'smart',

'snmp',

'sysctl',

'test',

'users',

'vlan',

'yumcmd']}

**[3] Show methods for a module (hiển thị phương thức cho một module)**

# show methods for "command" module

[root@certmaster ~]# func "minion01" call command list\_methods

{'minion01': ['run',

'config\_items',

'grep',

'exists',

'save\_config',

'module\_version',

'grep',

'list\_methods',

'module\_description',

'get\_method\_args',

'module\_api\_version']}

**Func – YumModule**

**Cách sử dụng cơ bản của YumModule.**

**[1]Chạy 'yum update' trên tất cả minions.**

( It's hadr to read them because "\n" is shown, replace it with sed 's/\\n/\n/g'.

[root@certmaster ~]# func "\*" call yumcmd update

('minion02', 'command: update \nkernel.x86\_64 0:2.6.32-71.24.1.el6 - i\

selinux-policy.noarch 0:3.7.19-54.el6\_0.5 - u\npolicycoreutils.x86\_64 0:2.0.83-19.8.e

l6\_0 - u\nkrb5-libs.x86\_64 0:1.8.2-3.el6\_0.7 - u\nglibc-common.x86\_64 0:2.12-1.7.el6\_0.5

u\nselinux-policy-targeted.noarch 0:3.7.19-54.el6\_0.5 - u\nglibc.x86\_64 0:2.12-1.7.el6

0.5 - u\nlogrotate.x86\_64 0:3.7.8-12.el6\_0.1 - u\npostfix.x86\_64 2:2.6.6-2.1.el6\_0 - u\n

tzdata.noarch 0:2011d-3.el6 - u\nkernel-firmware.noarch 0:2.6.32-71.24.1.el6 - u\ndhclie

nt.x86\_64 12:4.1.1-12.P1.el6\_0.4 - u\n') ('minion01', 'command: update \nselinux-policy.

noarch 0:3.7.19-54.el6\_0.5 - u\npolicycoreutils.x86\_64 0:2.0.83-19.8.el6\_0 - u\nkrb5-libs.

x86\_64 0:1.8.2-3.el6\_0.7 - u\nnss.x86\_64 0:3.12.8-3.el6\_0.0.sl6 - u\nglibc-common.x86\_6

4 0:2.12-1.7.el6\_0.5 - u\nselinux-policy-targeted.noarch 0:3.7.19-54.el6\_0.5 - u\nglibc.x8

6\_64 0:2.12-1.7.el6\_0.5 - u\nnss-sysinit.x86\_64 0:3.12.8-3.el6\_0.0.sl6 - u\nlogrotate.x86\_

64 0:3.7.8-12.el6\_0.1 - u\ntzdata.noarch 0:2011g-1.el6 - u\npostfix.x86\_64 2:2.6.6-2.1.el

6\_0 - u\ndhclient.x86\_64 12:4.1.1-12.P1.el6\_0.4 - u\nkernel.x86\_64 0:2.6.32-71.29.1.el6

- i\nkernel-firmware.noarch 0:2.6.32-71.29.1.el6 - u\n')

**Func – CommandModule**

**Cách sử dụng cơ bản của CommandModule. Cách này có thể thực hiện hầu hết các hoạt động với module này.**

**[1]**

**chạy 'yum update' trên tất cả minions.**

[root@certmaster ~]# func "\*" call command run "yum -y update" | sed 's/\\n/\n/g'

('minion02',

[0,

'Loaded plugins: fastestmirror

Loading mirror speeds from cached hostfile

\* base: ftp.jaist.ac.jp

\* extras: ftp.jaist.ac.jp

\* updates: ftp.jaist.ac.jp

Setting up Update Process

No Packages marked for Update

',

''])

('minion01',

[0,

'Loaded plugins: fastestmirror

Loading mirror speeds from cached hostfile

\* base: ftp.jaist.ac.jp

\* extras: ftp.jaist.ac.jp

\* updates: ftp.jaist.ac.jp

Setting up Update Process

No Packages marked for Update

',

''])

**[2]**

**thực thi 'ls -l /root' trên tất cả minions.**

[root@certmaster ~]# func "\*" call command run "ls -l /root" | sed 's/\\n/\n/g'

('minion02',

[0,

'total 20

-rw-------. 1 root root 1274 Jul 15 21:58 anaconda-ks.cfg

-rw-r--r--. 1 root root 8681 Jul 15 21:58 install.log

-rw-r--r--. 1 root root 3094 Jul 15 21:56 install.log.syslog',

''])

('minion01',

[0,

'total 20

-rw-------. 1 root root 1274 Jul 15 21:58 anaconda-ks.cfg

-rw-r--r--. 1 root root 8681 Jul 15 21:58 install.log

-rw-r--r--. 1 root root 3094 Jul 15 21:56 install.log.syslog

',

''])

**[3]**

**thực thi 'cat /etc/passwd' trên một minion.**

[root@certmaster ~]# func "minion01" call command run "cat /etc/passwd" | sed 's/\\n/\n/g'

('minion01',

[0,

'root:x:0:0:root:/root:/bin/bash

bin:x:1:1:bin:/bin:/sbin/nologin

daemon:x:2:2:daemon:/sbin:/sbin/nologin

adm:x:3:4:adm:/var/adm:/sbin/nologin

lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin

sync:x:5:0:sync:/sbin:/bin/sync

shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown

halt:x:7:0:halt:/sbin:/sbin/halt

mail:x:8:12:mail:/var/spool/mail:/sbin/nologin

uucp:x:10:14:uucp:/var/spool/uucp:/sbin/nologin

operator:x:11:0:operator:/root:/sbin/nologin

games:x:12:100:games:/usr/games:/sbin/nologin

gopher:x:13:30:gopher:/var/gopher:/sbin/nologin

ftp:x:14:50:FTP User:/var/ftp:/sbin/nologin

nobody:x:99:99:Nobody:/:/sbin/nologin

vcsa:x:69:69:virtual console memory owner:/dev:/sbin/nologin

saslauth:x:499:499:"Saslauthd user":/var/empty/saslauth:/sbin/nologin

postfix:x:89:89::/var/spool/postfix:/sbin/nologin

sshd:x:74:74:Privilege-separated SSH:/var/empty/sshd:/sbin/nologin

fermi:x:500:500::/home/fermi:/bin/bash

ntp:x:38:38::/etc/ntp:/sbin/nologin

dbus:x:81:81:System message bus:/:/sbin/nologin

haldaemon:x:68:68:HAL daemon:/:/sbin/nologin

smolt:x:498:498:Smolt:/usr/share/smolt:/sbin/nologin

**',**

**''])**

**[4]**

**thực thi 'touch test.txt' trên tất cả minion.**

[root@certmaster ~]# func "\*" call command run "touch /root/test.txt"

('minion02', [0, '', ''])

('minion01', [0, '', ''])

[root@certmaster ~]# func "\*" call command run "ls -l /root" | sed 's/\\n/\n/g'

('minion02',

[0,

'total 20

-rw-------. 1 root root 1274 Jul 15 21:58 anaconda-ks.cfg

-rw-r--r--. 1 root root 8681 Jul 15 21:58 install.log

-rw-r--r--. 1 root root 3094 Jul 15 21:56 install.log.syslog

-rw------- 1 root root 0 Jul 15 23:04 test.txt

',

''])

('minion01',

[0,

'total 20

-rw-------. 1 root root 1274 Jul 15 21:58 anaconda-ks.cfg

-rw-r--r--. 1 root root 8681 Jul 15 21:58 install.log

-rw-r--r--. 1 root root 3094 Jul 15 21:56 install.log.syslog

-rw------- 1 root root 0 Jul 15 23:04 test.txt

',

''])

**[5]**

**thực thi thay đổi chủ sở hữu và quyền sử hữu cho test.txt trên tất cả minion.**

[root@certmaster ~]# func "\*" call command run "chown nobody. /root/test.txt;chmod 600 /root/test.txt"

('minion02', [0, '', ''])

('minion01', [0, '', ''])

[root@certmaster ~]# func "\*" call command run "ls -l /root/test.txt" | sed 's/\\n/\n/g'

('minion02',

[0, '-rw------- 1 nobody nobody 0 May 14 23:04 /root/test.txt

', ''])

('minion01',

[0, '-rw------- 1 nobody nobody 0 May 14 23:04 /root/test.txt

', ''])

**[6]**

**khởi động lại ntpd trên tất cả minions.**

[root@certmaster ~]# func "\*" call command run "service ntpd restart" | sed 's/\\n/\n/g'

('minion02',

[0, 'Shutting down ntpd: [ OK ]

Starting ntpd: [ OK ]

', ''])

('minion01',

[0, 'Shutting down ntpd: [ OK ]

Starting ntpd: [ OK ]

', ''])

**Func – CopyFileModule**

**Cách sử dụng cở bản của CopyFileModule.Cách này hữu ích cho việc sao chép tệp vào minions.**

**[1]**

**Sao chép '/root/test2.txt' trên Certmaster cho tất cả minions**

[root@certmaster ~]# func "\*" copyfile -f /root/test2.txt --remotepath /root/test2.txt

[root@certmaster ~]# func "\*" call command run "ls -l /root" | sed 's/\\n/\n/g'

('minion02',

[0,

'total 24

-rw-------. 1 root root 1274 Jul 15 21:58 anaconda-ks.cfg

-rw-r--r--. 1 root root 8681 Jul 15 21:58 install.log

-rw-r--r--. 1 root root 3094 Jul 15 21:56 install.log.syslog

-rw-r--r-- 1 root root 89 Jul 15 23:14 test2.txt

-rw------- 1 nobody nobody 0 Jul 15 23:04 test.txt

',

''])

('minion01',

[0,

'total 24

-rw-------. 1 root root 1274 Jul 15 21:58 anaconda-ks.cfg

-rw-r--r--. 1 root root 8681 Jul 15 21:58 install.log

-rw-r--r--. 1 root root 3094 Jul 15 21:56 install.log.syslog

-rw-r--r-- 1 root root 89 Jul 15 23:14 test2.txt

-rw------- 1 nobody nobody 0 Jul 15 23:04 test.txt',''])