**percorso immagini e xml**

img: /var/lib/libvirt/images

xml: /etc/libvirt/qemu

password ubuntu: shadow

###################################################################################################################################

###################################################################################################################################

###################################################################################################################################

**modifiche su cyris.py:**

modificato il file cyris.py cambiando i tempi relativi all ssh

il file modules.py per il cambio dell'account fa riferimento al file add\_user.sh contenuto in instantiation/users\_managing

in questo file il comando è useradd .....

bisogna cambiarlo in adduser e invece di -s mettere --shell

risolvere la questione sulla password, non essendoci il comando -p

altrimenti in ubuntu la creazione dell'account non funziona correttamente

In fase di creazione del cyberrange viene richiesta la password per la connessione ssh, bisogna aggiungere all'interno dei py di cyris, prima di: ssh -o, la stringa: sspass -p theroot

in modo da avere: "sshpass -p theroot ssh -o ....."

>aggiunta all'interno di cyris.py relativo al settaggio del gateway, trovare altri moduli sui quali inserirla

la password viene anche chiesta durante la copia del file initif.conf, va aggiunto sshpass anche a scp

in particolare bisogna modificare in clone\_environment.py, funzione generate\_initif, mettendo:

command = "sshpass -p theroot scp -o UserKnownHostsFile=/dev/null -o StrictHostKeyChecking=no {0}initif.conf root@{1}:/bin/cyberrange/initif".format(self.directory, guest.getBasevmAddr())

VA FATTO SIA PER linux che per windows

inoltre, solo per windows, va aggiunto sshpass anche nella parte relativa al logout su cyris.py

###################################################################################################################################

###################################################################################################################################

###################################################################################################################################

**VIDEO**

modifiche al file instantiation/vm\_clone/vm\_clone\_xml.sh

cyris per questioni di prestazioni avvia i cloni con un'impostazione video limitata,

nel file ho commentato la parte presente relativa al video (sk video cirrus) e ho inserito:

echo " <video>" >> ${ABSPATH}images/${VM\_ID}\_config.xml;

echo " <model type='qxl' ram='65536' vram='65536' vgamem='16384' heads='1'/>" >> ${ABSPATH}images/${VM\_ID}\_config.xml;

echo " <address type='pci' domain='0x0000' bus='0x00' slot='0x02' function='0x0'/>" >> ${ABSPATH}images/${VM\_ID}\_config.xml;

echo " </video>" >> ${ABSPATH}images/${VM\_ID}\_config.xml;

MODIFICA 2:

Su windows ho installato i driver virtIO per migliorare le prestazioni, nell’xml da generare vanno fatte queste modifiche:

aggiungere un altro elemento “controller” sotto quello già esistente

echo " <controller type='virtio-serial' index='0'>" >> ${ABSPATH}images/${VM\_ID}\_config.xml;

echo " <address type='pci' domain='0x0000' bus='0x00' slot='0x05' function='0x0'/>" >> ${ABSPATH}images/${VM\_ID}\_config.xml;

echo " </controller>" >> ${ABSPATH}images/${VM\_ID}\_config.xml;

aggiungere l’elemento “channel” sotto </console>

echo " <channel type='spicevmc'>" >> ${ABSPATH}images/${VM\_ID}\_config.xml;

echo " <target type='virtio' name='com.redhat.spice.0'/>" >> ${ABSPATH}images/${VM\_ID}\_config.xml;

echo " <address type='virtio-serial' controller='0' bus='0' port='1'/>" >> ${ABSPATH}images/${VM\_ID}\_config.xml;

echo " </channel>" >> ${ABSPATH}images/${VM\_ID}\_config.xml;

commentare la riga “<graphic type” e aggiungere

echo " <graphics type='spice' autoport='yes'>" >> ${ABSPATH}images/${VM\_ID}\_config.xml;

echo " <image compression='off'/>" >> ${ABSPATH}images/${VM\_ID}\_config.xml;

echo " </graphics>" >> ${ABSPATH}images/${VM\_ID}\_config.xml;

in <memballoon>, <address type> cambiare slot=0x05 in slot 0x07

perchè 0x05 l’ho assegnato al controller

###################################################################################################################################

###################################################################################################################################

###################################################################################################################################

**SSH**

dare permessi a file known\_hosts:

>sudo chmod 777 /home/ubuntu/.ssh/known\_hosts

all'interno del file .../instantiation/sshkey\_hostmanem\_setup/sskey\_setup.sh commentare il percorso ssh-keygen e inserire questo:

>ssh-keygen -f "/root/.ssh/known\_hosts" -R ${vm\_addr} > /dev/null

Fare la stessa modifica anche per le macchine windows, quindi nel file: sshkey\_setup.sh

###################################################################################################################################

###################################################################################################################################

###################################################################################################################################

**DIMENSIONI VM**

per dare più spazio(es. 8gb) ad una vm:

sudo qemu-img resize /home/ubuntu/cyris/images/image.qcow2 +8G

avviare la macchina ed usare gparted per estendere il disco

###################################################################################################################################

###################################################################################################################################

###################################################################################################################################

**GESTIONE PLC**

i programmi plc vanno messi nella cartella ../models/nome\_modello/other\_files

se invece la creazione dello scenario avviene tramite interfaccia grafica allora il codice plc lo passa il front end.

per gestire questa situazione, nella funzione che gestisce lo yaml, l’argomento plc è opzionale, ovvero:

converter(yaml, codice, \*plcode)

questa modifica implica che quando vado a gestire plcode devo scrivere:

if plcode: ...codice

NOTA metto anche plcode = plcode[0]

perchè quando metto l’argomento opzionale, questo viene gestito come un oggetto (perchè potrebbe contenere vari input, quindi plcode[1], plcode[2], nel mio caspo è sempre [0] perchè gli passo solo un argomento, anche in presenza di più codici plc, passo cmq un solo dizionario che li contiene tutti)

**KALI**

scaricando kali light dovranno essere installati i tools di sicurezza separatamente, inserire il seguente repository all'interno del file

/etc/apt/source.list

inserire:

deb http://http.kali.org/kali kali-rolling main non-free contrib

da terminale digitare: apt-get update

per vedere la lista dei pacchetti installabili: apt-cache search kali-linux

installare il tool tra quelli presenti (es.top10): apt-get install kali-linux-top10

.

NElla parte relativa ad rc.local: in kali non è presente rc.local quindi si deve avviare lo script al boot in un altro modo

> mousepad /etc/crontab

inserire: @reboot /bin/cyberrange/initif/initif /bin/cyberrange/initif/initif.conf

e anche in un altro file, da terminale:

>crontab -e

si apre l'editor testuale e inserire ancora

@reboot /bin/cyberrange/initif/initif /bin/cyberrange/initif/initif.conf

.

per lo spegnimento della macchina:

impostazioni - gestione dell'energia - alla pressione del tasto di accensione: cambiare chiedi in spegni

TOR e PROXYCHAINS

bisogna anche installare:

apt-get install tor

apt-get install proxychains

###################################################################################################################################

###################################################################################################################################

###################################################################################################################################

**PREPARAZIONE IMMAGINE UBUNTU**

NOTA: negli xml associati alle immagini può esserci il campo uuid -> eliminarlo! altrimenti in fase di creazione del cyber range, non potranno essere create più macchine a partire da un'unica immagine

Cyris quando si avvia come prima operazione copia l'immagine disco specificata nel file YAML (dove c'è il percorso dell'XML associato all'immagine disco. NOTA: xml e immagine devono stare nella stessa cartella, devono avere stesso nome e il file immagine non deve avere estensione)

e la incolla nella cartella CyberRange e crea a quest'immagine un nuovo XML che differisce dall'originale per il MAC address e il nome dell'immagine

Poi avvia quest'immagine appena copiata. all'interno dell'immagine disco, preventivamente va messo uno script che sia avvia al boot che calcola l'IP dell'immagine a partire dal mac address, specificato nel file XML (in particolare l'ip sarà sempre 192.168.122.x dove x è la conversione decimale dell'ultimo numero esadecimale del macaddress esempio 6b diventa .107)

quest'operazione su ubuntu necessita di alcune modifiche per funzionare, in particolare bisogna:

accertarsi che il server SSH sia attivo all'avvio della macchina, in caso negativo digitare:

>sudo systemctl enable ssh

mettere bash invece di sh all'interno di rc.local

scaricare il pacchetto gawk

sostituire il comando chkconfig con update-rc.d all'interno dello script initif

PROCEDURA COMPLETA:

installare openssh-server

> sudo apt-get install openssh-server

installare il comando gawk:

> sudo apt-get install gawk

cambiare il nome dell'interfaccia di rete da ens a eth

> sudo gedit /etc/default/grub

incollare il testo 'net.ifnames=0 biosdevname=0' a GRUB\_CMDLINE\_LINUX in modo da avere:

GRUB\_CMDLINE\_LINUX="net.ifnames=0 biosdevname=0"

aggiornare grub:

> sudo update-grub

aprire il file interfaces

> sudo gedit /etc/network/interfaces

commentare TUTTO

abilitare il login ssh root

> sudo gedit /etc/ssh/sshd\_config

mettere yes su permitrootlogin: PermitRootLogin yes

usare theroot come password di amministratore

> sudo passwd root

digitare: theroot

copiare (via ssh) i file contenuti nella cartella di cyris: ..../cyris/instantiation/vm\_clone/initif (sono 3 file) in /bin/cyberrange/initif

creare quindi la cartella initif in bin/cyberrange

trasferire i file via ssh

> sudo scp initif root@ip\_macchina:/bin/cyberrange/initif

> sudo scp initif.conf root@ip\_macchina:/bin/cyberrange/initif

> sudo scp ip\_calc.sh root@ip\_macchina:/bin/cyberrange/initif

NOTA: se ci si prova a loggare ssh come root e inserendo la password c'è errore: permission denied, bisogna andare sulla macchina destinazione e inserire: sudo passwd root

e confermare 2 volte la password (cioè theroot)

dare permessi a questi 3 file

> sudo chmod 777 initif

> sudo chmod 777 initif.conf

> sudo chmod 777 ip\_calc.sh

aprire il file rc.local

> sudo gedit /etc/rc.local

cambiare alla prima riga /sh in /bash

aggiungere poi il percorso di 2 file copiati prima:

/bin/cyberrange/initif/initif /bin/cyberrange/initif/initif.conf;

rendere questo file eseguibile

> sudo chmod +x /etc/rc.local

cambiare il comando chkconfig (non riconosciuto in ubuntu) all'interno del file iniif

> sudo gedit /bin/cyberrange/initif/initif

cambiare

chkconfig NetworkManager off

in:

update-rc.d NetworkManager remove

disabilitare il networkmanager

> systemctl stop NetworkManager.service

> systemctl disable NetworkManager.service

eliminare la schermata di conferma spegimento sia nella lockscreen che una volta loggato (altrimenti cyris non può spegnere le macchine)

> gsettings set com.canonical.indicator.session suppress-logout-restart-shutdown true

per eliminare la conferma anche nella lockscreen, aprire

> sudo gedit /etc/acpi/powerbtn.sh

e aggiungere come primo comando:

/sbin/shutdown -h now

impostare le regole di firewall:

> su root

> iptables -P FORWARD DROP

> iptables-save > /bin/cyberrange/initif/iptables.conf

> gedit /etc/rc.local

aggiungere

> iptables-restore < /bin/cyberrange/initif/iptables.conf

**INSTALLAZIONE OPENMODELICA**

copiare/incollare l'indirizzo dell'installer di openmodelica dal sito ufficiale:

> for deb in deb deb-src; do echo "$deb http://build.openmodelica.org/apt `lsb\_release -cs` release"; done | sudo tee /etc/apt/sources.list.d/openmodelica.list

avviare l'update, se restituisce errore aggiungere: -allow-unauthenticated

> sudo apt update -allow-unauthenticated

installare:

> sudo apt install openmodelica

se restiruisce errore:

sudo apt-get update

sudo apt install openmodelica --allow-insecure-repositories

**FIREWALL**:

per la gestione del firewall, la macchina base ubuntu (che servirà per il firewall e per altre macchine) ha come policy: INPUT e OUTPUT su ACCEPT e FORWARD su DROP, in modo che soltanto le regole che metto nel file di configurazione siano consentite per l'inoltro dei pacchetti. Riavviando la macchina però la configurazione si perde e torna tutto su ACCEPT.

per risolvere, loggandomi come root sulla base image ubuntu ho salvato la configurazione giusta con:

=> iptables-save > /bin/cyberrange/initif/iptables.conf

e richiamato questa configurazione all'avvio della macchina, quindi:

=> sudo gedit /etc/rc.local

e aggiunto il comando (sotto a quelli gia presenti):

=> iptables-restore < /bin/cyberrange/initif/iptables.conf

in questo modo ogni macchina ubuntu al riavvio avrà FORWARD su DROP

alla fine di creazione del cyberrange, cyris scriverà all'interno della macchina (firewall) le regole del firewall impostate nel file di configurazione (che si aggiungeranno al DROP presente)

quindi, nel mio script post creazione, non devo indicare alcuna regola di iptables per nessuna delle macchine presenti

queste regole però si perderanno al successivo restart dello scenario, in particolare dovrò impostare sicuramente i gateway per tutte le macchine presenti, ma dovrò anche aggiungere le regole di firewall alla macchina firewall

a questo punto posso procedere in 2 modi:

1) nel restart indicare il gateway a tutte le macchine, prendere dal file setupfw.sh (nella cartella cyris/cyber range) le regole e scriverle nuovamente (non rilanciare solo il file perchè questo necessita di password ssh, quindi dovrei trovare un modo per passargli sshpass..)

2) alla fine della creazione del cyberrange, indico nel file script\_after\_clone di rifare nuovamente, e solo per la macchina firewall, iptables-save. IN questo modo poichè questo file viene caricato ad ogni riavvio avrò ancora la configurazione firewall

nel file script\_after\_clone\_restart devo solo rimettere i gateway e per la firewall solo abilitare il foward ipv4

NOTA: l'operazione di iptables-save la devo fare per ogni firewall presente nel cyber range

NOTA2: se una macchina ubuntu ha 2 interfacce collegate a 2 router diversi, se è una macchina ubuntu, questa avrà il FORWARD su DROP che in questo caso invece servirebbe su ACCEPT, gestire questo caso, magari abilitando per tutte le macchine che non siano firewall di essere sempre accept (agendo nello script after clone)

**PYTHON**

python risulta già installato, però scrivendo pip -V, risulta un'associazione a python2 invece che al 3.

disinstallare e reinstallare:

sudo apt-get remove python3-pip; sudo apt-get install python3-pip

nota:

installare modbus\_tk, opcua sia per l'utente normale che da root (cioè loggandosi con "su")

###################################################################################################################################

###################################################################################################################################

###################################################################################################################################

**PREPARAZIONE IMMAGINE WINDOWS**

Scaricare OpenSSH

32bits

https://github.com/PowerShell/Win32-OpenSSH/releases/download/0.0.24.0/OpenSSH-Win32.zip

64bits

https://github.com/PowerShell/Win32-OpenSSH/releases/download/0.0.24.0/OpenSSH-Win64.zip

Scaricare lo script:

https://github.com/motya1121/setup\_win\_openssh/blob/master/setup\_win\_openssh.ps1

oppure altro percorso per lo script: https://github.com/motya1121/setup\_win\_openssh

da qui scaricare l'intero pacchetto, il file che interessa è: "setup\_win\_openssh\_x64.ps1"

lanciando solo lo script non funziona quindi:

>Estrarre l'archivio OpenSSH-Win64.zip in C:\Programmi (ovvero C:\Program Files)

>Rinominare la cartella estratta OpenSSH-Win64 in OpenSSH

>Aprire come admin PowerShell ISE

>andare nella cartella di download dello script ps1 es: cd C:\Program Files\Users\winuser\Downloads

>eseguire nella shell: Set-ExecutionPolicy Unrestricted

>aprire il file ps1 (sempre all'interno di questo powershell ISE) e vedere la lista di comandi dello script che eseguiremo manualmente

> copiare singolarmente i comandi ed esegurli (tranne quelli di estrazione che sono gia stati fatti prima):

.\install-sshd.ps1

.\ssh-keygen.exe -A

.\FixHostFile......

.\netsh advFire....

.\Set-Service sshd ...

.\Set-Service ssh-agent...

Riavviare la macchina e provare il collegamento ssh

###################################################################################################################################

###################################################################################################################################

###################################################################################################################################

PROCEDURA X86:

prerequisiti: macchina x86 - Lingua Inglese - MOSTRA estensioni per file conosciuti

Per macchine 64bit utilizzare i relativi file contenuti nell'archivio

Utilizzare l'archivio contenente i file necessari: (sono per x86)

> Installare Net.Framework 4.6.1

> Installare PowerShell 5: decomprimere la cartella Win7-KB31911566-x86 e installare il relativo file (quello .msu)

> Decomprimere la cartella OpenSSH-Win32 e copiara in C:\Program files

> Rinominare questa cartella in OpenSSH

> Aprire Powershell come amministratore e digitare:

Set-ExecutionPolicy Unrestricted

Set-ExecutionPolicy Bypass

> Ritornare all'archivio con tutti i file, c'è "setup\_win\_openssh\_x86.ps1", lanciare questo file utilizzando powershell

> Riavviare la macchina e testare il collegamento ssh

> Copiare tutti i file contenuti nella cartella initif.01 e incollarli in C:\CyberRange\initif

(controllare che log.txt sia vuoto e che initif.conf abbia: eth0 192.168.122.0/24 ZZ:ZZ)

> aprire gpedit.msc, tab Powershell,

> aggiungere initif\_win.ps1 in modo da avere come percorso: C:\CyberRange\initif\initif\_win.ps1

> controllare l'ultimo ottetto del MAC address, riavviare la macchina e vedere se coincidono (convertito in decimale)

> per installare chocolatey aprire powershell come admin, andare sul sito indicato nella guida, nella pagina c'è un comando powershell, copiarlo ed eseguirlo in powershell per avviarel l'installazione

> start -tasto destro su "computer" "manage"

> andare su "Local Users and Groups" - "Users"

> cliccare tasto dx su administrator e rinominarlo in "root" e settare la password "theroot"

> bisogna abilitare il login come root: aprire come admin il prompt e digitare: net user root /active:yes

> loggarsi come root ed andare nella cartella: C:\Program Files\OpenSSH

> aprire con notepad il file sshd\_config e togliere il commento a PermitRootLogin

> testare la connessione ssh come root

NOTA: fatto questo vorrei mettere nuovamente su active:no la possibilità di loggarsi come root nella schermata di login, ma poi non funziona più il login root via ssh ...approfondire

Geckodriver lo metto nella cartella C:\CyberRange, già caricato sulla macchina

Riguardo la copia dei file sulla macchina via ssh, non riesco a far passare il comando ..root@192.x.x.x:'"percorso'"

questi doppi apici sono necessari se il percorso contiene una cartella con gli spazi, cioè come Program Files

per questo motivo copio tutto quello che serve in C:\CyberRange

e successivamente, con gli script\_after\_clone sposto dove serve. NOTA: in questo caso la sintassi dovrà essere esempio:

ssh root@192.x.x.x 'move C:\CyberRange\file "C:\Program Files\Destinazione"'

>>>>>guida installazione e script scadaBR su windows vedi file ScadaBR

installare python scaricando la versione 3.7 dal sito ufficiale (in fase di installazione selezionare anceh ADD PYTHON TO PATH)

per installare pip bisogna scaricare il file get-pip.py

spostarsi nella cartella dove è stato scaricato il py e lanciare: python get-pip.py

attraverso pip installare: pip install opcua pip install modbus\_tk pip install selenium

per installare geckodriver prima installare: visualc++, il pacchetto si chiama: vc\_redist.x86.exe oppure vc\_redist.x64.exe

geckodriver è scaricabile a: https://github.com/mozilla/geckodriver/releases

NOTA: faccio l'installazione sia per utente root che winuser

(scp su ubuntu windows funziona cosi: scp windows@ip:C/percorsofile /home/ubuntu oppure: scp windows@ip:C\\percorsofile /home/ubuntu)

le macchine windows non comunicano con le altre a causa della maschera di sottorete, che viene impostata a 255.0.0.0 invece che 255.255.255.0

la gestione di questi parametri avviene negli script initif caricati su windows, in particolare aprendo initif\_set\_win.ps1

si vede che avendo un prefix 8 mette 255.0.0.0, prefix 16 255.255.255.0, prefix 24 255.255.255.0

e in effetti cyris gli passa come parametro (vedendo initif.conf) l'ip e il prefix 8

quindi modificare il file initif.conf in cyris o lo script initif\_set\_win.ps1 mettendo 255.255.255.0 anche col prefix 8

netsh interface ipv4 set address name="Local Area Connection" source=static address=x.x.x.x mask=x.x.x.x gateway=x.x.x.x gwmetric=1

La strategia con windows è che alla prima creazione mi loggo ssh come normal user e faccio le operazioni, invece al restart devo necessariamente loggarmi come root per settare di nuovo i parametri di rete (il gateway)

FILE PRECARICATI SULL MACCHINA:

cartella initif: ubuntu, windows

file geckodriver: ubuntu su home/ubuntu ed è stato già configurato, windows su C:\CyberRange, richiamato nel file selenium

btn.png: ubuntu, windows, nella cartella di installazione upload di scadaBR

per far funzionare MODRSSIM su windows occorre installare C redistributable 2010 sia in versione x86 che x64

AVVIO DEGLI SCRIPT SU WINDOWS

su windows devo lanciare separatamente gli script, script1 come utente root e script2 come utente normale (a differenza di linux dopo posso indicare l'user/root stesso da riga di comando)

a seconda del modeler scelto, script1 e script2 conterranno comandi diversi

MIGLIORAMENTO PERFORMANCE

per migliorare le performace bisogna installare i driver virtio in windows:

scaricare l’iso con i driver dal link:

<https://fedorapeople.org/groups/virt/virtio-win/direct-downloads/stable-virtio/virtio-win.iso>

in kvm aggiungere un drive cd caricando l’iso scaricata:

* cliccare su “aggiungi hardware”
* sezione “storage”
* selezionare “select or create custom storage”
* cliccare su “manage” e caricare l’iso scaricata
* tipo dispositivo: CD-ROM device

all’avvio della macchina ci sarà il drive ottico virtuale con all’interno l’iso scaricata

andare in “manage” di windows, sezione “Device Manager”

cercare PCI DEVICE e cliccare su “Update Driver Software” cercando i file all’interno del drive ottico virtuale.

Il driver da installare, nel caso di windows7 x64 si trova in: D:\Balloon\w7\amd64

al termine effettuare un reboot.

**INSTALLARE SU WIN ANCHE il file spice-guest-tool e riavviare**

questa modifica oltre ad installare vari driver, correggerà anche i problemi col mouse senza dover quindi installare il touchpad usb in kvm

DISCO VIRTIO

è possibile migliorare le prestazioni passando dal disco IDE al tipo VIRTIO.

per farlo è necessario fare alcune operazioni.

* a macchina spenta, cliccare su “aggiungi hardware” - “storage”
* “create a disk image for virtual machine” mettendo ad esempio 2g
* come tipo selezionare “Disk Device” e come BUS “VirtIO”
* avviare la vm, e andare in “manage” - “Device Manager” - “disk device”
* compariranno 2 dischi tra cui quello VirtIO
* spegnere la macchina, eliminare il disco VirtIO aggiunto (quello da 2gb)
* cambiare il tipo disco del disco principale passando da IDE a VIRTIO
* avviare la macchina, che chiederà un restart per effettuare le modifiche

Una soluzione migliore sarebbe creare la VM di windows da zero utilizzando i driver già in fase di creazione (da testare)

link di riferimento:

https://access.redhat.com/documentation/en-us/red\_hat\_enterprise\_linux/6/html/virtualization\_host\_configuration\_and\_guest\_installation\_guide/form-virtualization\_host\_configuration\_and\_guest\_installation\_guide-para\_virtualized\_drivers-mounting\_the\_image\_with\_virt\_manager