Cách thức sử dụng chroot cho application để chống hack

Vấn đề hệ thống chạy trên môi trường internet hay bị các hacker tấn công xâp nhập qua các lỗ hổng của webserver (apache, nginx, tomcat,…) để xâm nhập vào hệ thống tạo các script đào coi hoặc thả virus lên Server gây lỗi hệ thống; Để giải quyết vấn đề này, cách dễ nhất là dùng docker; còn nếu không đóng gói đc application vào docker thì có thể sử dụng cách thức chroot; Bản chất chroot là một sanbox đơn giản nguyên thủy (thuật ngữ sanbox dùng trong bảo mật là để ngăn cách application khỏi các thứ # và nó sẽ tan vỡ ngay sau khi app stop)

Để giải quyết vấn đề này; trước tiên bắt buộc phải sử dụng một user khác không có quyền root để chạy ứng dụng; Nguyên lý thao tác của nó như sau:

## Tạo một phân vùng riêng hoặc thư mục dành riêng cho chroot

thuật ngữ jail dùng để chỉ định một thư mục với đúng nghĩa đen của nó – nhà tù cho ứng dụng – nó dùng để cách lý và chỉ cho app hoạt động trong đó – kiểu giống như đóng nó vào container của docker vậy;

Nếu có thể sử dụng một phân vùng riêng biệt với các tùy chọn nosuid, nodev và noexec cho thư mục chroot là tốt nhất; Điều này sẽ cải thiện an ninh do người dùng sẽ không thể chạy kích hoạt chương trình SUID và tập tin thiết bị bên trong một jail: Thực hiện = cách mount nó lên với các option đó:

Ví dụ mount trong fstab: UUID=0aef28b9-3d11-4ab4-a0d4-d53d7b4d3aa4 /tmp ext4 defaults,nodev,nosuid,noexec 1 2

Hoặc trực tiếp bằng lênh mount:

mount -o rw,noexec,nosuid,nodev,bind /tmp/ /var/tmp/

Ở đây tốt nhất là cho nó vào fstab để boot lên nó tự mount chứ không nên dừng lệnh mount – vì khi restart nó sẽ mất;

## Tạo các thư mục và copy thư viện cần thiết vào jail

Trước tiên chúng ta cần xác định xem application của chúng ta chạy thế nào và cần đóng gói cái gì vào jail; ví dụ muốn đóng gói /bin/bash vào jail thì cần phải tạo thư mục /bin trong jail và copy /bin/bash vào trong bin của jail:

ví dụ:

mkdir -p /chroot/app/

mkdir -p /chroot/app/dev

mknod /chroot/app/dev/null c 1 3

mknod /chroot/app/dev/random c 1 8

mknod /chroot/app/dev/urandom c 1 9

mkdir -p /chroot/app/bin

cp /bin/bash /chroot/app/bin/bash

### Đóng gói các thư viện đi kèm vào jail

Lưu ý ở đây nếu chỉ copy cái /bin/bash vào thì nó không chạy được vì vẫn thiếu các thư viện đi kèm của bash;

vì vậy cần phải phân tích và copy các thư viện phụ trợ của nó vào trong jail thì mới chạy được;

Linux hỗ trợ cho chúng ta công cụ lld để tìm kiếm dependency; cách thứ sử dụng như sau:

[root@chatbotdev tmp1]# ldd /usr/lib/jvm/java-1.8.0-openjdk-1.8.0.352.b08-2.el7\_9.x86\_64/jre/bin/java

linux-vdso.so.1 => (0x00007ffeebfd2000)

libpthread.so.0 => /lib64/libpthread.so.0 (0x00007f4eacd91000)

libz.so.1 => /lib64/libz.so.1 (0x00007f4eacb7b000)

libjli.so => /usr/lib/jvm/java-1.8.0-openjdk-1.8.0.352.b08-2.el7\_9.x86\_64/jre/bin/../lib/amd64/jli/libjli.so (0x00007f4eac96c000)

libdl.so.2 => /lib64/libdl.so.2 (0x00007f4eac768000)

libc.so.6 => /lib64/libc.so.6 (0x00007f4eac39a000)

/lib64/ld-linux-x86-64.so.2 (0x00007f4ead1af000)

[root@chatbotdev tmp1]#

* Vậy chỉ cần copy các thư viện này vào tên thư mục tương ứng trong jail là được;

Như vậy muốn đưa cái gì vào jail thì hãy đi copy các thứ dependency của nó vào là được;

### Chạy ứng dụng trong jail

để chạy ứng dụng trong jail thì cần sử dụng một câu lệnh đặc biệt:

chroot <new root path> <command to run> <argument(s)>

ví dụ:

su - appgw

chroot /app/appgw /app/appgw/testing/testing.sh start

Đối với các phần mềm đóng gói sẵn như NgiNX hoặc apache thì nó đã có sẵn chế độ config cho chroot rồi; ngoài ra yum cũng cho phép ta cài package vào jail chroot như sau:

yum install -y httpd --installroot=/chroot/app

ở đây option --installroot sẽ chỉ định là cài đặt vào jail là =/chroot/app

Chroot Apache trên Centos 7

Ví dụ cài đặt vào jail =/chroot/httpd như sau:

# mkdir -p /chroot/httpd/dev

# yum install -y httpd --installroot=/chroot/httpd

# cd /chroot/httpd/dev

# rm -f null

# mknod /chroot/httpd/dev/null c 1 3

# mknod /chroot/httpd/dev/random c 1 8

# mknod /chroot/httpd/dev/urandom c 1 9

# ls -al

total 0

drwxr-xr-x. 2 root root 47 Sep 2 19:26 .

dr-xr-xr-x. 17 root root 224 Sep 2 19:17 ..

crw-r--r--. 1 root root 1, 3 Sep 2 19:26 null

crw-r--r--. 1 root root 1, 8 Sep 2 19:26 random

crw-r--r--. 1 root root 1, 9 Sep 2 19:26 urandom

## Tạo trang testing default:

# cd /chroot/httpd/var/www/html/

# cat >> index.html << EOF

> <html>

> <head>

> <body>

> <h1>Apache Homepage from Chroot Jail.</h1>

> </body>

> </html>

> EOF

## Add firewall enable port http

# firewall-cmd --permanent --add-service=http

success

# firewall-cmd --reload

success

## Tạo service cho systemctl

#cp /chroot/httpd/etc/sysconfig/httpd /etc/sysconfig/

#cp /chroot/httpd/usr/lib/systemd/system/httpd.service /usr/lib/systemd/system/

#vi /usr/lib/systemd/system/httpd.service

Type=simple

ExecStart=/sbin/chroot /chroot/httpd /usr/sbin/httpd $OPTIONS -DFOREGROUND

ExecReload=/sbin/chroot /chroot/httpd /usr/sbin/httpd $OPTIONS -k graceful

## Start service

# systemctl daemon-reload

# systemctl start httpd

# systemctl enable httpd

Chroot Apache 2 Web Server trên Red Hat/CentOS5/6

**Sandbox có thể là một công cụ hữu dụng cho việc trợ giúp bảo vệ hệ thống chống lại sự thỏa hiệp, miễn là bạn sử dụng nó sáng suốt.** Trong bài này sẽ giới thiệu cho các bạn một Sandbox cụ thể, chroot, đây là một Sandbox đơn giản, dễ sử dụng, cũng như khả năng có sẵn cao.

Trong thuật ngữ bảo mật ứng dụng, sandbox là một cơ chế kỹ thuật mà với cơ chế đó, một số hoạt động của các ứng dụng có thể được cô lập với phần còn lại của hệ thống theo một cách nào đó, hy vọng nó sẽ bảo vệ hệ thống của bạn chống lại được bất cứ ác tâm hay những điều không may có thể rơi vào các ứng dụng đó.

Tiện ích chroot thường là một cách đơn giản nhất, nhanh nhất và có thể nói là dễ dàng nhất cho việc sandbox một ứng dụng nào đó – đặc biệt một máy chủ - nơi vấn đề bảo mật là quan trọng hàng đầu.

Khi sử dụng chroot, một mẩu phần mềm sẽ được khởi chạy bên trong một kiểu cấu trúc thư mục root lỗi, theo lý thuyết là làm cho phần mềm bị “đánh lừa” và nghĩ rằng không có gì bên ngoài cấu trúc thư mục đó, vì vậy sẽ cách ly phần còn lại của hệ thống với bất cứ vấn đề có tiềm ẩn sự ảnh hưởng gây hại trong việc chạy phần mềm này: Việc này sẽ giúp tránh được tấn công thay đổi các thành phần của thư mục gốc trên linux.

Bất cứ quá trình nào được chroot, root đều tránh được sự truy nhập trực tiếp vào cấu trúc thư mục root lỗi của nó, tuy nhiên vẫn tiềm ẩn một số ứng dụng có thể chạy phần mềm khác và được lợi dụng để tạo các quá trình có thể phá hủy môi trường chroot. Phương pháp kinh điển để phá hủy sandbox đó là tạo khả năng có thể gắn lại (remount) thư mục hiện hành.

Trong phạm vi tài liệu này, tôi sẽ hướng dẫn các vấn đề sau:

* Đảm bảo một máy chủ web Apache 2 trên Red Hat Enterprise Linux/CentOS Linux dùng mod\_chroot
* Cấu hình virtual host trong chroot jail.
* Xử lý sự cố chroot jail Apache.

## Yêu cầu

* Server: Apache 2 web server.
* Thư mục Jail: /httpdjail là thư mục đóng băng chroot đến đó;
* User/Group: apache/apache (không bao giờ chạy chroot sử dụng user root).
* Thư mục tên miền ảo cho tất cả các miền trong jail: /home/httpd.
* PHP được cấu hình thông qua mod\_php mặc định.
* Hướng dẫn được thử nghiệm trên CentOS/RHEL 6.x.

## Thêm thông tin về thư mục Jail: / httpdjail Tạo một thư mục jail như sau:

# J=/httpdjail

# mkdir $J

* Không tạo thư mục /dev bên trong jail.
* Không tạo tập tin thiết bị đặc biệt bên trong jail.
* Đừng sao chép shell hoặc bất kỳ tập tin thực thi duy nhất khác vào bên trong jail.
* Không chạy httpd hoặc php/perl/python với user root.
* Nếu có thể gắn /httpdjail sử dụng một phân vùng riêng biệt với các tùy chọn nosuid, nodev và noexec. Điều này sẽ cải thiện an ninh do người dùng sẽ không thể chạy kích hoạt chương trình SUID và tập tin thiết bị bên trong một jail: Thực hiện = cách mount nó lên với các option đó:

Ví dụ mount trong fstab: UUID=0aef28b9-3d11-4ab4-a0d4-d53d7b4d3aa4 /tmp ext4 defaults,nodev,nosuid,noexec 1 2

Hoặc trực tiếp bằng lênh mount:

mount -o rw,noexec,nosuid,nodev,bind /tmp/ /var/tmp/

## **Cài đặt Apache, PHP và MySQL** Cài đặt gói cần sử dụng lệnh yum:

yum install mysql mysql-server httpd php-mysql php-pear php-xml php-mysql php-cli php-imap php-gd php-pdo php-devel php-mbstring php-common php-ldap php httpd-devel

Bây giờ, tạo ra các thư mục cần thiết bên trong jail:

Mã:

mkdir -p $J/var/run

chown -R root.root $J/var/run

mkdir -p $J/home/httpd

mkdir -p $J/var/www/html

mkdir -p $J/tmp

chmod 1777 $J/tmp

mkdir -p $J/var/lib/php/session

chown root.apache $J/var/lib/php/session

* $J/var/run sẽ lưu trữ PID và các tập tin khác.
* $J/var/lib/php/session đường dẫn tập tin session trong PHP (cấu hình trong php.ini).
* $J/tmp - Được sử dụng bởi nhiều script và phần mềm cms để tải lên tập tin.

## Cài đặt mod\_chroot

mod\_chroot làm cho Apache chạy trong một môi trường chroot an toàn dễ dàng. Bạn không cần phải tạo ra một hệ thống phân cấp đặc biệt có chứa thư mục /dev, /lib, /etc.  
  
mod\_chroot cho phép chạy Apache trong một nhà tù chroot không có tập tin bổ sung. chroot() system call được thực hiện vào cuối thủ tục khởi động - khi tất cả các thư viện được nạp và các tập tin log mở. Nếu bản apache chưa có mod\_chroot thì có thể download source về và compile như sau:

cd / opt / à -->Điều hướng đến thư mục lưu cài đặt

wget [http](https://mangmaytinh.net/tags/http/)://core.segfault.pl/~hobbit/mod\_chroot/dist/mod\_chroot-0.5.tar.gz -> Tải về mã nguồn

tar -zxvf mod\_chroot-0.5.tar.gz à -->Giải nén tập tin tải về

cd mod\_chroot-0.5 --> Điều hướng đến thư mục giải nén

apxs -cia mod\_chroot.c -->Chạy lệnh cài đặt

**Cấu hình Apache mod\_chroot**  
Mởi tập tin /etc/httpd/conf/httpd.conf. Gõ lệnh:

C=/etc/httpd/conf/httpd.conf

vi $C

Thiết lập đường dẫn PidFile trong đó các máy chủ nên ghi lại mã số quá trình của nó khi nó bắt đầu. Tìm dòng mà đọc như sau:

PidFile run/httpd.pid

Thay bằng:

PidFile /var/run/httpd.pid

Tiếp theo, thêm chỉ thị ChrootDir, hãy nhập:

ChrootDir /httpdjail

Tìm dòng như sau:

ServerRoot "/etc/httpd"

Thêm các dòng sau:

LockFile /var/run/httpd.lock

CoreDumpDirectory /var/run

ScoreBoardFile /var/run/httpd.scoreboard

Hãy chắc chắn rằng dòng mod\_chroot.so tồn tại. Ví dụ, Linux 64 bit nên có dòng như sau:

LoadModule chroot\_module /usr/lib64/httpd/modules/mod\_chroot.so

Dòng cấu hình Linux 32 bit:

LoadModule chroot\_module /usr/lib/httpd/modules/mod\_chroot.so

Lưu lại và đóng tập tin.  
  
**Vô hiệu hóa SELinux cho Apache**  
Cần phải vô hiệu hóa SELinux cho apache, hãy nhập:

Mã:

setsebool httpd\_disable\_trans 1

Patch /etc/init.d/httpd  
Mở tập tin /etc/init.d/httpd, hãy nhập:

vi /etc/init.d/httpd

Tìm dòng như sau:

# Start httpd in the C locale by default.

HTTPD\_LANG=${HTTPD\_LANG-"C"}

Thêm dòng sau đây (thiết lập ROOT đến $J):

ROOT=/httpdjail

Tìm dòng stop() như sau:

stop() {

echo -n $"Stopping $prog: "

killproc -d 10 $httpd

RETVAL=$?

echo

[ $RETVAL = 0 ] && rm -f ${lockfile} ${pidfile}

}

Thay thế nó như sau (cần phải liên kết /var/run/httpd.pid đến $j/var/run/httpd.pid, vì thế hoạt động stop sẽ làm việc):

stop() {

/bin/ln -s $ROOT/var/run/httpd.pid /var/run/httpd.pid

echo -n $"Stopping $prog: "

killproc -d 10 $httpd

RETVAL=$?

echo

[ $RETVAL = 0 ] && rm -f ${lockfile} ${pidfile}

}

Lưu lại và đóng tập tin. Thiết lập quyền không thay đổi vào tập tin /etc/init.d/httpd để tập tin không thể được sửa đổi, cập nhật bởi yum, xóa hay đổi tên, không có liên kết có thể được tạo ra đến tập tin này và không có dữ liệu có thể được ghi vào tập tin. Chỉ có superuser hoặc một quá trình sở hữu những khả năng CAP\_LINUX\_IMMUTABLE mới có thể thiết lập hoặc xóa thuộc tính này:

chattr +i /etc/init.d/httpd

Làm thế nào để start httpd chroot?

Gõ lệnh sau đây:

/etc/init.d/httpd start

Bạn không thấy bất kỳ lỗi gì trong tập tin /var/log/httpd/error\_log:  
Ví dụ kết quả đầu ra:

[Sun Dec 21 18:43:09 2008] [notice] core dump file size limit raised to 18446744073709551615 bytes

[Sun Dec 21 18:43:09 2008] [notice] SELinux policy enabled; httpd running as context root:system\_r:initrc\_t

[Sun Dec 21 18:43:09 2008] [notice] suEXEC mechanism enabled (wrapper: /usr/sbin/suexec)

[Sun Dec 21 18:43:09 2008] [notice] Digest: generating secret for digest authentication ...

[Sun Dec 21 18:43:09 2008] [notice] Digest: done

[Sun Dec 21 18:43:10 2008] [notice] mod\_chroot: changed root to /httpdjail.

[Sun Dec 21 18:43:10 2008] [notice] Apache/2.2.3 ([CentOS](https://mangmaytinh.net/tags/centos-8/)) configured -- resuming normal operations

Làm thế nào để stop httpd chroot?

/etc/init.d/httpd stop

Làm thế nào để khởi động lại httpd chroot?

/etc/init.d/httpd restart