Github地址：<https://github.com/Tripwire/tripwire-open-source>

官网：https://www.tripwire.com/

Tripwire

1. 介绍

Tripwire是一款最为常用的开放[源码](http://www.21tx.com/src/)的完整性检查工具，它生成目标文件的校验和并周期性的检查文件是否被更改

1. 运行原理

Tripwire可以对要求校验的系统文件进行类似md5的运行，而生成一个唯一的标识，即“快照”snapshot。当这些系统文件的大小、inode号、权限、时间等任意属性被修改后，再次运行Tripwire，其会进行前后属性的对比，并生成相关的详细报告。

1. 组成

Tripwire主要由策略和数据库组成。策略不仅指出Tripwire应检测的对象即文件和目录，而且还规定了用于鉴定违规行为的规则。一般情况下，对于/root、/bin和/lib目录及其中文件的任何修改都应视为违规行为。数据库则用来存放策略中规定的检测对象的快照。只要建立了策略和数据库，我们就可以随时用快照来比较当前的文件系统，然后生成一个完整性检测报告，从而判断系统的完整性是否受到攻击。除了策略和数据库外，Tripwire还有一个配置文件，用以控制数据库、策略文件和Tripwire可执行程序的位置等。

1. 功能

可以对linux的文件进行一个监控，基于tripwire数据库中存放的文件数据，当进行检查的时候，发现linux中的任意文件发生变化，都能够识别。

1. 配置文件介绍

twcfg.txt 文件确定 Tripwire 使用的变量（例如 tripwire 报告文件的位置、e-mail 地址、报告级别）

twpol.txt 策略文件告诉 Tripwire 监视什么样的文件以及忽略什么文件。

1. 站点（site）密钥和本地（local）密钥介绍

本地密钥用于数据库文件，站点密钥用于配置文件和策略文件。

1. 优缺点

优点：文件发生的任何细微变化，都能被tripwire检查到，并生成详细报告工管理员辨别。可自定义所检查的文件夹以及文件以及定义文件风险级别，当发生变化通过邮件报警。

缺点：如果策略文件中设定检查的文件过多，再执行全盘检查的时候会占用系统资源，过程缓慢。

1. 安装

1.yum源

Centos6

wget -O /etc/yum.repos.d/epel.repo <http://mirrors.aliyun.com/repo/epel-6.repo>

Centos 7

wget -O /etc/yum.repos.d/epel.repo <http://mirrors.aliyun.com/repo/epel-7.repo>

yum clean all

2.安装

yum -y install tripwire

**3.密码设置操作**

vim /etc/tripwire/twinstall.sh

*DIR=/etc/tripwire*

*SITE\_KEY=$DIR/site.key*

*LOCAL\_KEY=$DIR/`hostname`-local.key*

*#创建site和local密钥，这时会要求输入口令；如果这两个密钥业已存在，则可以跳过此步骤。其中，site密钥存放在site.key文件中，而local密钥则*

*存放在hostname-local.key*

*twadmin --generate-keys --site-keyfile $SITE\_KEY*

*twadmin --generate-keys --local-keyfile $LOCAL\_KEY*

*#利用site密钥对默认配置文件twcfg.txt进行签名，并将签名（而非被签名的文件twcfg.txt）存放于文件tw.cfg之中。*

*#利用site密钥对默认策略文件twcfg.txt进行签名，并将签名（而非被签名的文件twcfg.txt）存放于文件tw.pol之中。*

*twadmin --create-cfgfile --cfgfile $DIR/tw.cfg --site-keyfile $SITE\_KEY $DIR/twcfg.txt*

*twadmin --create-polfile --cfgfile $DIR/tw.cfg --site-keyfile $SITE\_KEY $DIR/twpol.txt*

*cd $DIR*

*#设置权限*

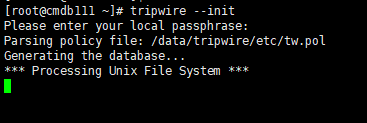
*chown root:root $SITE\_KEY $LOCAL\_KEY tw.cfg tw.pol*

*chmod 600 $SITE\_KEY $LOCAL\_KEY tw.cfg tw.pol*

**4、初始化数据库**

tripwire –init

要求输入local passphrase



**5、根据初始配置文件进行全面检查**

tripwire –check > /data/no-directory.txt

备注：会有好多报错，因为初始策略中存在系统没有的文件。对此进行筛选

vim /etc/tripwire/twsed.sh

*#!/bin/bash*

*for f in `sed 's:\/:\\\/:g' /data/no-directory.txt.bak |grep "Filename"|awk -F: '{print $2}'`*

*do*

*#echo $f*

*sed -i "s:$f:#$f:g" /etc/tripwire/twpol.txt*

*done*

**6.对策略文件进行重新签名**

twadmin -m P /etc/tripwire/twpol.txt

**7.重新初始化数据库**

tripwire --init

**8.进行全面检查**

tripwire --check

**9.根据要求订制我们需要扫描的文件夹以及文件，并设置警告级别。**

规则样式

(

Rulename = “规则名”,

)

{

/data -> $(SEC\_READONLY) ;

}

配置文件规则含义

*# -忽略以下属性*

*# +检查以下属性*

*# a访问时间戳(与+ CMSH互斥)*

*# b分配的块数*

*# c inode创建/修改时间戳*

*# d inode所在设备的ID*

*# g所有者的组ID*

*# I inode号码*

*# l增长文件(例如日志文件)*

*# m 修改时间戳*

*# n个链接*

*# p权限和文件模式位*

*# r inode指向的设备ID(仅对设备对象有效)*

*# s文件大小*

*# t文件类型*

*# u所有者的用户ID*

*# C crc-32哈希*

*# H HAVAL哈希*

*# M MD5哈希*

*# S SHA哈希*

*SEC\_CRIT = $(IgnoreNone)-SHa;不能更改的关键文件*

*SEC\_SUID = $(IgnoreNone)-SHa;设置了SUID或SGID标志的#二进制文件*

*SEC\_BIN = $(只读);不应该改变的二进制文件*

*SEC\_CONFIG = $(Dynamic);不经常更改但经常访问的配置文件*

*SEC\_LOG = $(正在增长);#文件会增长，但不应该改变所有权*

*sec\_constant = +tpug;永远不应该更改权限或所有权的目录*

*SIG\_LOW = 33;对安全影响最小的非关键文件*

*SIG\_MED = 66;具有重大安全影响的非关键文件*

*SIG\_HI = 100;#关键文件，这是重要的漏洞点*

1. 下载最新的源码
2. wget <https://github.com/Tripwire/tripwire-open-source/archive/master.zip>
3. 安装依赖

yum -y install gcc\* openssl

1. 配置

./configure –prefix=/data/tripwire

1. 编译安装

make

make install

注意：期间需要设定site passphrase与local passphrase

备注：若未设置密码执行一下命令创建心得密码

twadmin --generate-keys --site-keyfile $SITE\_KEY

twadmin --generate-keys --local-keyfile $LOCAL\_KEY

1. 安装部署完成

配置文件：/data/tripwire/etc/twcfg.txt

策略文件：/data/tripwire/etc/twpol.txt

1. 初始化tripwire数据库

/data/tripwire/sbin/tripwire –init

您将被问到“local-key”密码短语，您可能会收到错误消息“no such directory”

得到错误是因为系统没有在tripwire配置中已经定义的目录和文件。 为了解决这个错误，我们需要编辑tripwire配置'twpol.txt'并重新签署tripwire配置。

1. 从tripwire生成日志错误

tripwire --check | grep Filename > /data/no-directory.txt

1. 用脚本调整tripwire策略文件

vim /data/tripwire.sh

for f in $(grep "Filename:" /data/no-directory.txt | cut -f2 -d:); do  
sed -i "s|\($f\) |#\\1|g" /data/tripwire/etc/twpol.txt  
done

1. 手动增加所要监控的文件夹以及忽略监控的文件夹。

10、用twadmin命令重新生成并重新签署tripwire策略

twadmin -m P /data/tripwire/etc/twpol.txt

输入您的“网站密钥”密码

11、初始化数据库

/data/tripwire/sbin/tripwire --init

输入您的“本地密钥”密码

12、根据策略文件进行全盘检查

/data/tripwire/sbin/tripwire –check

13、结合计划任务每天进行检查

注解：

SITEKEYFILE为远程密钥，LOCALKEYFILE用来加密数据库的本地密钥。LOOSEDIRECTORYCHECKING设置为ture时会去掉一些相同报告。SYSLOGREPORTING默认不作日志记录。

/data/tripwire/sbin/twprint -m r -twrfile <reportfile.twr> 打印报告

/data/tripwire/sbin/tripwire --check 根据规则文件进行全面检查

相关连接

<https://blog.csdn.net/ytfhjhv/article/details/87450602>规则配置文件中详细的预定义变量介绍

Inotify-tools

wget -O /etc/yum.repos.d/epel.repo http://mirrors.aliyun.com/repo/epel-6.repo

/proc/sys/fs/inotify/max\_queued\_evnets  
表示调用inotify\_init时分配给inotify instance中可排队的event的数目的最大值，超出这个值的事件被丢弃，但会触发IN\_Q\_OVERFLOW事件。  
/proc/sys/fs/inotify/max\_user\_instances  
表示每一个real user ID可创建的inotify instatnces的数量上限。  
/proc/sys/fs/inotify/max\_user\_watches  
表示每个inotify instatnces可监控的最大目录数量。如果监控的文件数目巨大，需要根据情况，适当增加此值的大小，例如：  
echo 30000000 > /proc/sys/fs/inotify/max\_user\_watches

inotifywait相关参数  
Inotifywait是一个监控等待事件，可以配合shell脚本使用它，下面介绍一下常用的一些参数：  
λ -m， 即--monitor，表示始终保持事件监听状态。  
λ -r， 即--recursive，表示递归查询目录。  
λ -q， 即--quiet，表示打印出监控事件。  
λ -e， 即--event，通过此参数可以指定要监控的事件，常见的事件有modify、delete、create、attrib等。

Rpm-Vf 针对安装的rpm包的配置文件等检测。

RPM安装方式快速部署tripwire