Linux 定时器及其在网络安全中的应用

林绍太　张会汀　（广州 暨南大学 信息科学技术学院 510632）

摘要：定时器是 Linux 操作系统内核的核心机制之一，所有与时间相关的进程都要用到定时器，并保证任务的准时调度。本文在分析定时器的实现原理及 TCP/IP 定时器的基础上，对 Linux 定时器在网络安全中的应用做出了具体的分析。

关键词：Linux　定时器　IP 欺骗　SYN Flood

1　引言

定时器(timer)是 Linux 中提供的一种定时服务的机制。它所起的作用是在某个特定的时间唤醒某个进程来做一些工作。

大多数现代计算机系统都有二个时钟源，分别叫做 RTC(Real Time Clock)和 OS 时钟。RTC 也称为硬件时钟，由 PC 主板上依靠电池供电的晶振来提供。OS 时钟建立在 RTC 上，初始化以后将完全由操作系统控制。OS 时钟不仅可以提供可靠的系统当前时间，还可以提供精确的时间间隔。

Linux 内核中定时器是一个很重要的部分，所有与时间相关的进程都要用到定时器，一个定时器设置不好将会影响到系统整个的定时器系统。因此本文将首先简要的介绍定时器的实现机制，然后再介绍具体的应用。

2　Linux 定时器的实现原理

操作系统应该能够在将来某个时刻准时调度某个任务。所以需要一种能保证任务较准时被调度运行的机制。希望支持某种操作系统的微处理器必须具有一个可周期性中断的可编程间隔定时器。这个周期性中断被称为系统时钟滴答，它象节拍器一样来组织系统任务。

Linux 的时钟观念很简单：它表示系统启动后的以时钟滴答记数的时间。所有的系统时间都基于这种量度，它和系统中的一个全局变量 jiffies 的名称相同。

Linux 包含两种类型的系统定时器，它们都可以把将在某个系统时间上被调用的例程进行排列，但是它们的实现稍有区别。

第一种是老的定时器机制——Timer_table，用一指向 timer_struct 数据结构的 32 位指针的静态数组以及当前活动定时器的屏蔽码：time_active(一长整型变量)来实现。此定时器表中的位置是静态定义的(类似 bottom half 的处理表 bh_base)。其入口在系统初始化时被加入到表中。第二种是相对较新的定时器，它使用一个以升序的到期时间排列的 timer_list 结构链表。timer_list 结构是为了弥补 timer_table 只有 32 个 timer 的不足而创建的。一般来说，现在 timer_table 中的只有一些系统定时器，而用户的定时器都在 timer_list 中，timer_list 的链表结构允许有无限多个 timer，方便了用户。

这两种方法都使用 jiffies 作为终结时间，这样，希望运行 5 秒的定时器将不得不将 5 秒时间转换成 jiffies 的单位，并且将它和以 jiffies 记数的当前系统时间相加，从而得到定时器的终结时间。在每个系统时钟滴答时，定时器的 bottom half 处理过程被标记成活动状态以便调度器在下次运行时能进行系统定时器队列的处理。定时器的 bottom half 处理过程会处理两种类型的系统定时器。老的系统定时器将检查 timer_active 位是否置位。

如果活动定时器已经到期则其定时器例程将被调用，同时 bottom half 中相应的活动位也被清除。新定时器位于 timer_list 结构链表中的入口也将受到检查。每个过期定时器将从链表中清除，同时它的例程将被调用。新定时器机制的优点之一是能传递一个参数给定时器例程。

3　TCP/IP 与定时器

TCP 是建立在不可靠 IP 基础上的面向连接的传输控制协议。由于采用了超时重传和拥塞控制(滑动窗口)技术，因此 TCP 具有很高的可靠性，但这种可靠性与定时器提供的准确定时是分不开的。在 Linux 的 TCP/IP 协议中，最常用的定时器如下：

3.1　连接建立定时器 (connect timer)

在连接建立阶段，当发送了 SYN 包后，就启动连接定时器。如果在 75 秒内没有收到应答，则放弃连接建立。

3.2　重传计时器(retransmit timer)

为了控制丢失的或丢弃的提出报文段，TCP 使用处理对报文段确认等待时间的重传计时器。当 TCP 发送报文段时，它就创建该特定报文段的重传计时器。若在规定的时间内，发送方还没有收到确认，则定时器被激活，函数 tcp_re-

stransmit_timer 开始执行，它动态调整 RTT(Round Trip Time)的值并重发报文。

3.3 坚持计时器(persist timer)

在连接的一方需要发送数据但对方已通告窗口大小为 0 时，就需要设置 TCP 坚持定时器。发送方使用坚持定时器来周期性地向接收方查询，一旦发现接受方窗口通知值非零，则发送方又回到传输状态，开始正常的数据传输。

3.4 保活定时器(keep alive timer)

保活定时器用来防止两个 TCP 之间连接处理时间长时期的空闲。在连接空闲两个小时后，在一个连接上发送一个探测报文段来完成保活功能。若发送了 10 个探测报文段(每一个相隔 75 秒)还没有响应，就假设客户出了故障，并终止该连接。

4 IP 欺骗攻击与定时器

IP 欺骗就是伪造数据包源 IP 地址的攻击，其实现的可能性基于两个前提：第一，目前的 TCP/IP 网络在路由数据包时，只判断目的 IP 地址，并不对源 IP 地址进行判断，这就给伪造 IP 创造条件；第二，两台主机之间，存在基于 IP 地址的认证授权访问，这就给会话劫持创造了条件。其结果是未授权的远端用户进入目标主机系统。

在 IP 欺骗中，存在三个角色：一是攻击目标 S，二是被目标主机信任的一台主机 C，三是入侵者 I。I 必须执行两步操作：首先，与 S 建立一个虚假连接；然后，阻止 C 向 S 报告网络证实系统的问题。主机 I 必须假造 C 的 IP 地址，从而使 S 相信从 I 发来的包的确是从 C 发来的。

主机 I 伪造 IP 地址步骤如下：

I→S：SYN(序列号＝M)，SRC＝C

S→C：SYN(序列号＝N)，ACK(应答号＝M＋1)

I→S：ACK(应答号＝N＋1)，SRC＝C

其中，A 的 ISN(N)是 X 估算出来的。同时，主机 I 应该阻止主机 C 响应主机 S 的包。为此，I 可以等到主机 C 因某种原因终止运行，或者阻塞主机 C 的操作系统协议部分，使它不能响应主机 S。

如前所述，TCP 维持一个连接建立定时器。因些在 75 秒之内服务器端口将无法对其他 TCP 连接作出反应，这样通过大量的 SYN 类型数据包将阻塞主机 C 的端口。这就是所谓的 SYN Flood 攻击。在主机系统中，抵御 SYN Flood 攻击可以采用下列措施：

- 增加 TCP 监听套接字未完成连接队列的最大长度；
- 减少未完成连接队列的超时等待时间；
- 使用诸如 SYN Cookies 这样的特殊措施。

入侵者还可以利用虚假状态转移来长时间阻塞主机 C 的一个网络端口。比如说从 SYN－RCVD 到 CLOSE－WAIT 的状态转移，实现如下：

I→C：SYN FIN 同时置位(序列号＝N)

C→I：ACK(应答号＝N＋1)

(1) 从主机 I 到主机 C 发送一个带有 SYN 与 FIN 标志位同时置位的 TCP 包。

(2) 假设主机 C 首先处理 SYN 标志，生成一个带有相应 ACK 标志位置位的包，并使状态转移到 SYN－RCVD，然后处理 FIN 标志，使状态转移到 CLOSE－WAIT，并向 I 回送 ACK 包。

(3) 主机 I 不向主机 C 发送其他任何包。主机的 TCP 机将固定在 CLOSE－WAIT 状态。如果保活定时器特征被使用，一般 2 个小时后 TCP 将会重置并转移到 CLOSED 状态。

为了防止从 SYN－RCVD 到 CLOSE－WAIT 状态的伪转移，需要改变操作系统中 TCP 操作的部分相关代码，使得当 TCP 机处于 SYN－RCVD 状态时，忽略任何对等主机发来的 FIN 包。在基于 Linux 防火墙系统的具体应用中，还可以考虑禁止 TCP 状态图的所有外部转移，并根据实际情况对保活定时器参数进行设置。

5 小结

定时器是 Linux 操作系统内核的核心机制之一，所有与时间相关的进程都要用到定时器，并保证任务的准时调度。本文在分析定时器的工作原理及 TCP/IP 定时器的基础上，针对防范 IP 欺骗攻击中 Linux 定时器的应用做出了具体的分析并提出了解决的方法。

参考文献

1 Gary R. Wright, W. Richard Stevens TCP/IP 详解卷 2，谢希仁译，机械工业出版社，2000.7。

2 博嘉科技，Linux 防火墙技术探秘，国防工业出版社，2002.10。

3 张耀疆，聚集黑客，人民邮电出版社，2002.9。