

Diffserv 中带宽代理 BB 的实现

邱瑜 朱淼良

(浙江大学计算机系 浙江大学网络中心 杭州310027)

Bandwidth Resource Dynamic Allocation in Differentiated Services Model

QIU Yu ZHU Miao-Liang

(Department of Computer, Network Center, Zhejiang University, Hangzhou 310027)

Abstract To employ Differentiated Services, efficient and flexible resource allocation mechanism is needed. Static bandwidth resource allocation is not suitable for the situations in which traffic varies greatly with times. If customers bought bandwidth according to their highest traffic demands, some resource will be wasted when traffic load is light. On the other hand, the static allocated bandwidth may not satisfy the growing traffic demands. So it is necessary to allocate bandwidth resource in dynamic way. We bring up a prototype of BB, discussing the necessary components to realize the dynamic bandwidth resource allocation and further more, our BB can provides advanced reservation for bandwidth resource. Simulation tests show our design is feasible.

Keywords Diffserv, BB, Resource allocation

1 引言

区分服务(Differentiated Services, Diffserv)模型^[1]是一种新的具有 QoS 能力的 Internet 服务模型。它把服务分成为数有限的几类,然后在区分服务区域的边缘节点上,对业务流进行分类、标记、整形、计量,而在其中心节点,只对分类后的汇聚流进行快速传递,因而保证了中心节点的简单性。和 Interserv 服务模型相比, Diffserv 模型更易于实现,更具有可扩展性,极有可能成为下一代的 Internet 服务模型^[2]。

在区分服务中,区分服务的提供者和接受者之间(一个 Diffserv 区域和另一个 Diffserv 之间或者一个客户和一个 ISP 之间)要签订 SLA 协定(服务等级协定)。SLA 规定数据流的业务类别和能得到的网络带宽资源。SLA 协定可以是静态的,由网络管理人员事先商量好后手工进行设定,在较长的一段时间内都维持不变。但是,很显然静态的分配方式不够灵活。假如用户的带宽需求大部分时间里都小于其订购的带宽,那么用户的利益受到了伤害。在业务量变化剧烈的网络中,静态的资源分配方式要么造成带宽浪费,要么不能及时满足变化的带宽需求。静态分配带宽也不能满足一些如 IP 电话等的按 per call 收费的商业模式。因此,要使 Diffserv 服务模型在实际中得到广泛的应用,必须实现动态的 SLA 协定,也即动态带宽分配。而且,带宽的分配管理应当是两维的,不但要有数量上的分割,还应当有时间段上的分割。只有这样,才能满足那些需要事先预留带宽资源的应用,比如为了确保一个 Internet 上的网络视频会议有足够的带宽,会议安排人员可能需要提前预定带宽。

IETF 的区分服务工作组提出利用 BB 来实现带宽资源分配^[3,4],但并没有提及如何具体实现一个 BB。本文给出一个能够提供资源预留、有动态分配带宽资源能力的 BB 的原型实现框架,并进行了仿真试验。

2 基于 BB 的资源分配模型

BB 是一个 DS 区域内的带宽资源管理 Agent。它拥有区域内各节点的带宽资源信息和路由消息。BB 负责维护本地和周边其它区域签订的 SLA 协议,接受本地或其它 DS 区域的带宽分配请求,根据网络服务策略和当前的带宽资源分配情况来管理和分配带宽资源,配置边缘路由器。另外,它负责传递区域间的带宽分配请求消息,以此完成边缘-边缘的带宽订购。

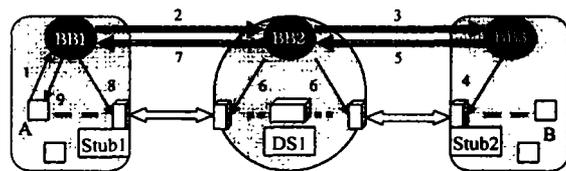


图1 区分服务的动态资源分配模型

基于 BB 的动态资源分配模型如图1所示。图1中有两个端网络 Stub1、Stub2,它们通过区分服务区域 DS1 相连接。假设 Stub1 中的主机节点 A 要与 Stub2 中的主机节点 B 通信,它先进行带宽资源订购。一个即时的带宽订购过程如下所述:

1. Stub1 中的主机节点 A 向本地带宽代理 BB1 发出资源分配请求(RAR);
2. 假如 BB1 接受请求,它根据请求的目的地 B,向 DS1 的 BB2 发出请求;
3. 假如 BB2 接受请求,它把带宽分配请求发往 BB3;
4. 假如 BB3 接受请求,它对数据流要进入的边缘路由器进行配置,更新本地资源信息;
5. BB3 向 BB2 返回接受请求的消息;
6. BB2 配置数据流要经过的入口路由器和出口路由器;
7. BB2 向 BB1 返回接受请求的消息;

邱瑜 博士研究生,研究方向:智能 agent、网络 QoS 控制。朱淼良 博士生导师,研究方向:机器人、计算机视觉、计算机网络。

- 8. BB1配置数据流的出口路由器;
- 9. BB1向主机节点 A 返回接受带宽分配请求的消息;

这样,通过相应路径上的 BB 的串连工作,完成了边缘到边缘的带宽资源订购,主机节点 A 就能以订购的服务级别,享受带宽服务。相反,如果某个 BB 发现本地资源不能满足要求,就向上游的 BB 发回拒绝服务的消息,直至该消息返回给主机节点 A。A 收到拒绝服务的消息后,可能降低服务级别或者另择时间发送请求。

考虑到区分服务提供的是一种优质服务,有理由相信用户为了在特定时间段内得到高级别的服务,会有提前预订带宽资源的需求。对于预定带宽资源的情况,资源的分配分成两步。第一步是发出资源预留请求。每个 BB 在接受该类请求后,和即时订购一样,在数据流要经过的每个 DS 区域,BB 查看资源能否满足要求,但当请求被接受的消息返回时,BB 只保留资源的预订信息,但并不马上配置边缘路由器。而是等到数据流要实际发送的那一刻,资源预定者再向 BB 发出要使用资源的请求,BB 核对其请求后,只需沿途配置每个边缘路由器。之后,预定者就可以发送数据。

一般来说,核心路由器比边缘路由器具有高得多的速度,但如果入口路由器的数据流集中流向某个核心路由器时,也会发生堵塞。因此有必要监视核心路由器的资源使用情况。我们在 BB 中维护一个网络核心链路带宽使用情况的数据库。在接受一个带宽资源分配请求时,BB 更新从入口路由器到出口路由器所经过的每一条链路的带宽使用总量,如果该值已经达到了规定的上限,BB 将拒绝新的资源分配请求。而核心路由器本身并不需要 BB 的配置,也不对经过的数据流进行监视。

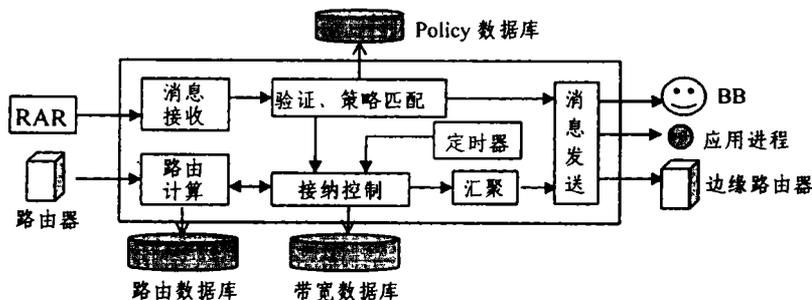


图2 BB 的结构

3 BB 的实现

3.1 BB 的框架结构

根据前面所述的资源分配模型,我们提出一个 BB 的原型实现框架,它包括如图2所示的功能模块:

3.1.1 消息接收、发送模块 消息接收模块负责接收本地或其它 DS 区域的和资源分配有关的消息,主要包括资源分配请求消息及其应答消息。消息发送模块发送的消息包括三类:一是 BB 与其它 DS 区域 BB 之间传递的消息;二是 BB 向应用进程反馈的资源分配请求的结果,这类消息一般只有在端网络中才有;三是 BB 配置边缘路由器的消息。

3.1.2 验证、策略匹配模块 当 BB 的消息接收模块收到一个资源分配请求(RAR)后,首先要对消息的发送者身份进行合法性验证,并检查该请求是否符合网络服务策略,如服务等级、服务时间段、带宽要求有没有超出限定等。如果通不过,则直接由消息发送模块向请求者发出拒绝服务的消息。否则由接纳控制模块计算现有的网络资源能否满足请求。

3.1.3 接纳控制模块 接纳控制模块是 BB 的核心模块,它拥有区域内各路由设备的当前带宽资源使用情况。当有一个新的分配请求到来时,它利用路由计算模块获得业务流要经过的路由设备,计算在每个路由设备上是否有足够的带宽资源能满足分配请求。如果不能满足,BB 发出拒绝服务的消息。如果能满足且请求的目的地在本地,BB 对边缘路由器进行配置,并且发出接受请求的消息。如果本地资源能满足要求但请求的目的地不在本地,则该类消息将被汇聚然后送往其下游 DS 区域的 BB。

服务级别。

3.2 带宽分配

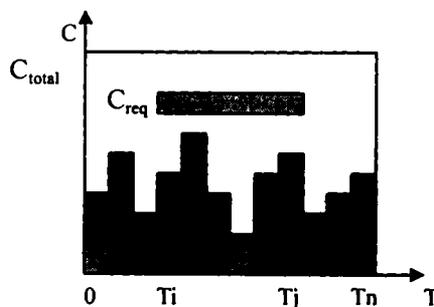


图3 带宽资源的分配

为了满足对带宽资源的预订要求,带宽使用情况的表述方式应该是二维的,即应当知道带宽在某个时间点或时间段上的量的分配。如图3,假设一个路由器对某种服务可以提供

3.1.4 汇聚模块 它对发往其它 DS 区域的带宽资源分配请求进行汇总,即根据所有的请求,统计它们对各相邻区域的总的带宽请求,然后汇聚后的带宽请求被送到其它 BB。汇聚的主要目的是减轻 BB 的负担,以免过多的消息收、发和路由器配置行为。

3.1.5 路由计算模块 其作用是根据请求消息 RAR 中规定的流的源地址和目的地址,计算流在本地所要经过的入口路由器、内部路由器和出口路由器。接口控制模块据此来计算相应路由器上的资源能否满足请求。路由计算模块要完成该功能,必须要动态地知道本地网络的拓扑情况。其方法有两种:一是由路由计算模块向路由器进行查询;二是像路由器那样,接收每个路由器定时广播的路由消息,然后计算路由。很明显,后者不会增加路由器的负担,更有灵活性。

3.1.6 定时器 其作用是查看对一个用户的服务时间是否已经超过了 RAR 请求的服务时间期限。当服务时间到时,BB 会重新配置路由器,不再提供服务给该用户或者降低

的带宽容量为 C_{total} , 横坐标是时间单位。这样带宽的使用情况可以按时间间隔 Δt 离散化表示成 $(C_0, C_1, \dots, C_n), C_m (m=1, \dots, n)$ 表示在第 m 个时间点已经被使用或预定的带宽资源数量。这样, 对于 BB 来说, 当收到一个资源分配请求 $RAR(Src, Dst, Class, C_{req}, Start_Time, End_Time)$ 时, 先用 $Start_Time$ 和 End_Time 求得对应的离散时间段集, 如 T_i 到 T_j 段, 然后查看 $\min(C_{total} - C_m) (m=i, \dots, j)$ 是否大于 C_{req} 。如果是, 则可以分配带宽。如果不是, 则表明没有足够的带宽供分配。

3.3 资源订购协议

为了实现资源订购, 我们设计了如表1所示的最小资源订购协议集, 用于客户端向 BB 发出资源使用请求。

表1 资源订购协议

名称	协议内容	返回值
即时资源订购请求	Do-Resv, Src, Dst, Class, Last-Time	Reserved-ID
资源预订请求	Advanced-Resv, Src, Dst, Class, Start-Time, End-Time	Reserved-ID
预订资源使用申请	Use-AdvancedResv, Reserved-ID	Success/Failure
取消资源订购	Cancel-Resv, Reserved-ID	
资源续订请求	Prolong-Resv, ReservedID, Prolong-Time	Success/Failure

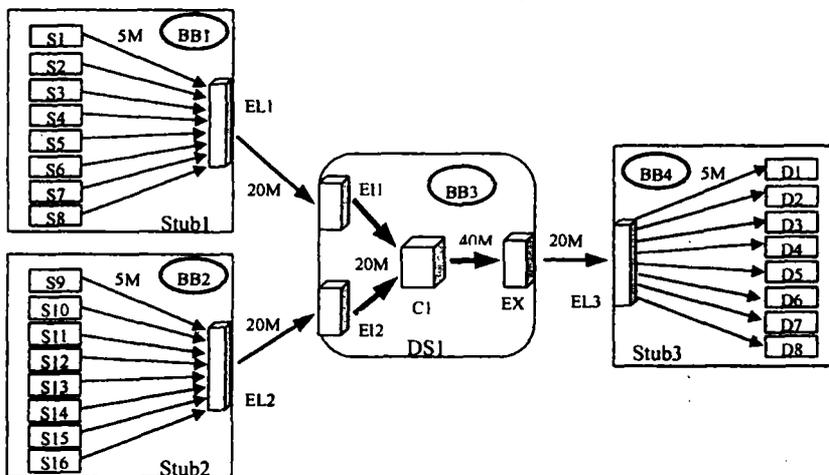


图4 仿真试验的网络拓扑图

4.2 PHB 的实现

在仿真试验中, 每个路由器有三个队列, 分别和 EF、AF 和 BE 三种 PHB 服务相对应。

根据 EF PHB 的定义, 在每个路由节点上, 汇聚流的到达速率不大于汇聚流的离开速率, 因此一个 EF 队列的平均长度很小或为零。EF PHB 服务有低丢失率、低延迟和低畸变的特点, 可以实现高水平的 QoS 服务, 如虚拟专线服务。为了实现 EF PHB, 我们严格限制业务流的发送速率, 当一个业务流超过它带宽分配请求中规定的速率时, 数据包将被丢弃。

AF (Assured Forward) PHB 和 EF PHB 不同, AF PHB 的实现目的是保证用户即使在拥塞情况下也能得到一定的带宽资源, 但包的延迟和抖动并没有保证。试验中我们使用具有 IN/OUT 的随机早期检测算法 (RIO) 实现 AF PHB。当流进入路由器时, 预定带宽以内的流被标记为 IN 流, 超出部分被标记为 OUT 流, 它们以不同的 DSCP 域值来表示。IN 流和

4. 仿真试验

根据前面所述的资源动态分配模型, 我们进行了仿真试验, 考察方案的可行性。仿真试验是在加州大学 Berkeley 分校的 NS 仿真器^[5]基础上开发实现的。该仿真软件包含有 Nortel 公司^[6]提供的实现 AF PHB 的 Diffserv 代码。我们对代码进行改造, 实现了三种典型的 PHB: EF^[7]、AF^[8] 和 BE。在此基础上, 我们按前面所述的框架结构实现具体的 BB, 来进行带宽资源的动态分配仿真。

仿真试验是在 PII200, 128M 内存的机器上实现的, 操作系统为 Linux6. 2。

4.1 网络拓扑

仿真试验的网络拓扑如图4。一共有四个自治域, Stub1、Stub2和 Stub3是能够使用区分服务的端网络, EL1、EL2、EL3 都是叶节点边缘路由器, 它们和最终的带宽消费节点相连, 端节点 $S_i (i=1, \dots, 16)$ 和边缘路由器之间的带宽为 5M。在这些路由器上, 每个用户流被单独监视。DS1 是一个具有完全区分服务能力的传输网络, 它由一个核心路由器和三个边缘路由器组成。由端网络进入的汇聚的业务流在入口路由器 E11、E12 上被监视, 以防它们违反 SLA 协定, EX 路由器则对流出的业务流按其和 Stub3 之间的 SLA 协定进行整形。E11、E12 与核心路由器 C1 之间的带宽为 20M, C1 到 EX 的带宽为 40M。每个网络上都有一个 BB, 负责带宽资源的分配。

OUT 流位于同一物理队列的两个不同的虚拟队列中, 但 IN 流的丢失率低于 OUT 流的丢失率。拥塞发生时, 属于 OUT 流的包被大量丢失。

BE (Best-effort Forward) PHB 即当前 Internet 上对数据包转发的行为方式。对业务流不提供任何的 QoS 保证。

我们假定在每个路由器上, EF、AF 和 BE 类服务的带宽资源分配比例为 25%、25%、50%。因此, 对 EF、AF 类服务最多可以各分配到 5M 带宽。由于这两种服务都有一定的 QoS 保证, 所以使用前必须先申请资源, 而 BE 类服务则用不着向 BB 请求资源。

在队列的调度上, 我们采用 WRR (Weighted Round-robin) 算法, EF、AF、BE 队列的权值分别为 7、1、2, 按此权值, 每个边缘路由器最高可以传递速率为 14M/s 的 EF 类业务流, 保证流的流入速率不大于离开速率。

4.3 试验

4.3.1 带宽资源的动态请求试验 我们先按照表1进行即时的带宽分配请求和数据发送试验。其中 BE 类服务不需要进行带宽分配,其它类服务在节点发出请求并得到允许后发出数据流。

表2 带宽资源分配请求和数据流的发送

序号	时间	源地址-目的地址	开始时间	结束时间	PHB	要求带宽	实际发送速率	流量特性	请求结果
1	0	S ₁ -D ₁ I=1,2,3	0	80	BE		4M	持续	
		S ₁ -D ₁ I=4,5	0	80	EF	2M	2M	持续	Accept
		S ₆ -D ₆	0	60	AF	2M	2M	持续	Accept
		S ₁ -D ₁ I=7,8	0	80	AF	1M	1M	持续	Accept
2	20	S ₉ -D ₄	20	100	EF	3M		持续	Deny
		S ₁₀ -D ₆	20	100	AF	3M		持续	Deny
2	30	S ₁₈ -D ₃	30	80	BE		2M	间隔	
		S ₁₄ -D ₅	30	80	EF	1M	1M	间隔	Accept
		S ₁₅ -D ₇	30	80	EF	1M	1M	间隔	Accept

从表2中可以看出,在0时刻,Stub1中的节点 S₁(i=1...8)要和 Stub4中的节点 D₁(i=1...8)进行通信。S₄、S₅发出 EF 类的带宽资源分配请求,S₆、S₇、S₈发出 AF 类的带宽资源分配请求。它们要求的带宽数量都没有超出 EF、AF 类服务的最高限额,因此请求被接受。EF 类服务、AF 类服务的带宽要求各为4M,实际发送的数据速率也为4M,BE 类服务实际的数据发送速率为12M,因此带宽需求量总和等于沿途的 EL1、EI1、EX、EL3 路由器所能提供的最大带宽供应能力。在第20秒,S₉、S₁₀发出带宽分配请求,它们的请求在 EL2、EI2 上都能被接受,但在 EX 上,EF、AF 类的带宽资源只剩下2M,无法满足要求,因此请求被拒绝。在第30秒时,S₁₄和 S₁₅发出新的请求,对 EF 类、AF 类服务的带宽需求都为1M。当请求被接纳后,S₁₄、S₁₅和 S₁₈开始发送数据。但它们的数据流和 S₁-S₈节点的固定速率持续发送的流不同,是指数分布的,以一定的速率发送100ms,然后中止500ms。当它们都在发送数据时,在 EL3 路由器上,EF、AF 类业务流对带宽的占用将达到各自的带宽上限。而进入 EL3 的 BE 类流最高的带宽数为14M,因此总的带宽需求将超出 EL3 的带宽能力,EL3 必然会拥塞。

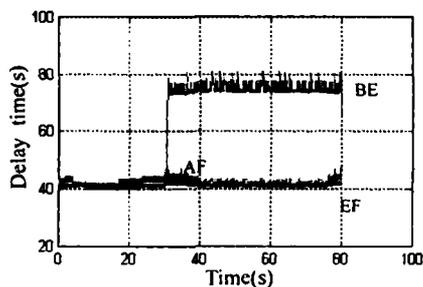


图5 数据包的延迟

图5是动态分配资源情况下各类数据包在0~80秒内的延迟时间曲线。在前30秒内,由于数据流量没有超出各个路由器的传递能力,三种数据包的延迟时间都在40ms左右。而在30秒后,由于流量负荷的增加,路由器 EL3 上产生拥塞现象,BE 类数据包的延迟时间增加到70多毫秒,大致在75~82ms 间上下波动。而 EF 类和 AF 类的数据包延迟时间基本上不变,这是因为 BB 能够准确实现动态的资源分配,保证高等级服务的接入不会超出带宽资源的上限,从而满足业务流的 QoS 特

性。

4.3.2 资源的预订和分配试验 图6和图7分别是 Stub1中节点 S₁和 Stub2中节点 S₉对 EF、AF 类服务的带宽需求变化图。对带宽的分配有两种方式可以选择,一是在0时刻就各个时间段的带宽需求进行资源预订;二是到带宽需求变化前的一刻进行即时的资源分配请求。在此,我们假设 EF 类服务很关键,要确保能得到服务,因此在0时刻就一次性预订好。而 AF 类服务相对次要些,就在使用前请求资源,只要确保使用时有一定的带宽。

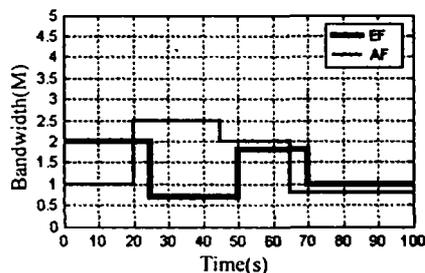


图6 S₁节点的带宽需求量变化

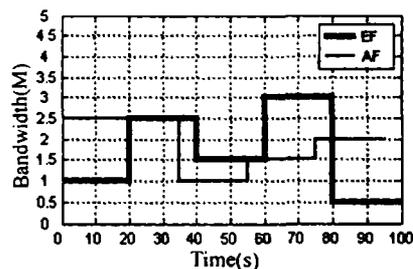


图7 S₉节点的带宽需求变化

以 S₁为例,根据其带宽资源需求变化,对资源的请求如表3。在0时刻,请求1-4完成全部的 EF 类服务的带宽分配,2、3、4属于资源预留的请求。然后在带宽需求变化的时间点上,请求7、9、11发出,要求使用预留的带宽资源,沿途的 BB 设置好路由器后,S₁就可以发送数据。

表3 S₁节点的带宽分配及使用

序号	时间	开始时间	结束时间	PHB	要求带宽	注释
1	0	0	25	EF	2M	
2	0	25	50	EF	0.7M	预留
3	0	50	70	EF	1.8M	预留
4	0	70	100	EF	1M	预留
5	0	0	20	AF	1M	
6	20	20	45	AF	2.5M	
7	25	25	50	EF	0.7M	使用预留
8	45	45	65	AF	2M	
9	50	50	70	EF	1.9M	使用预留
10	65	65	100	AF	0.8M	
11	70	70	100	EF	1M	使用预留

按照图6、7的带宽需求变化情况,在0时刻,当 S₁和 S₉发出分配请求后,路由器 EX 的带宽资源分配情况如图8、9所示。EF 类的带宽由于采用预订的方式,因此在整个时间段上都有分配,而 AF 类的带宽分配只有0~35这一段时间内。

(下转第64页)

有效控制,则这段恶意代码的破坏将屏蔽,进入 MC 状态,直至该代码自动终止,返回 G 状态(但目前尚未发现能够对拥有管理员权限的恶意进程进行有效控制的方法)。

(6)状态 TR:如果攻击者进程的运行被容侵系统检测到了,则系统进入了 TR 状态。任务管理器可以立即终止该程序,使系统返回到 G 状态。

(7)状态 GD:如果终止该程序失败,系统还可以尝试提供降级服务。降级的方法可以视攻击者进程所造成的破坏而定,进入 GD 状态。然后,等待管理员修复系统,再返回 G 状态。

(8)状态 FS:如果攻击者进程已经造成了严重损害,无法继续提供 Internet 服务,则服务器可以尝试进入 FS 状态,关闭所有的网络服务,关闭 IIS 服务器。等待管理员修复系统,再返回 G 状态。

(9)状态 F:如果服务器无法自动关闭,不能进行自我保护,则应立即报警,请求管理员处理,进入 F 状态。

在这个实例中,我们可以看到尽管 MC 状态的实施方法尚未成熟,但从总体上,状态转换模型描述这个容侵实例的工作流程。

结论 本文在介绍了 Firewall、IDS 和 IT 之后,引入了状态转换模型,对容侵系统的工作流程进行抽象的描述。本文对状态转换图中的状态进行了详细的解释,并通过实例说明了容侵系统在抵抗入侵的时候,系统状态在图中的迁移过程和相应方法。

在一般的容侵实例中,未必状态转换模型中的所有状态、

事件和处理方法都有所对应。在有些情况下可能只存在状态转换图中的某些状态,也有可能某些恢复方法找不到对应的实施手段。但从整体上讲,状态转换模型对于描述一般容侵系统的工作流程,是很有代表性的。

在状态转换模型中,我们需要进一步研究的课题还有:对协同/并行攻击的容侵模型的建模与分析、容侵系统与 COTS 服务器的组织结构、对于特定环境的重组和恢复方法、入侵检测方法等。

参考文献

- 1 Characterizing Intrusion Tolerant Systems Using A State Transition Model. <http://www.anr.mcnc.org/projects/SITAR/papers/darpa00.pdf>, 2000
- 2 Bugtraq 1600. <http://www.securityfocus.com/bid/1600>, 2000
- 3 Auscert advisory aa-2000.02. <http://ftp.auscert.org/pub/auscert/advisory/AA-2000.02>, 2000
- 4 Du W, Mathur A P. Vulnerability testing of software system using fault injection. [Technical Report Coast TR-98-02]. Department of Computer Science, Purdue University, 1998
- 5 MCNC, D. University. Sitar: A scalable intrusion tolerant architecture for distributed services. [Technical report, Research Proposal to DARPA BAA-00-15]. 2000
- 6 Lee P A, Anderson T. Fault Tolerance: Principles and Practice. Springer Verlag, 1990
- 7 Amoroso E G. Intrusion Detection: An Introduction to Internet Surveillance, Correlation, Trace Back, Traps, and Response. Intrusion. Net Books, 1999
- 8 Krsul I, Spa@ord E H, Tripunitara M V. Computer vulnerability analysis. [Technical Report Coast TR 98-07]. Department of Computer Science, Purdue University, 1998
- 9 Ellison R J, et al. Survivability: Protecting your critical systems. IEEE Internet Computing, 1999, 3(6): 55~63
- 10 Northcutt S, Novak J. Network Intrusion Detection: An Analysts' Handbook. New Riders, Sep. 2000

(上接第50页)

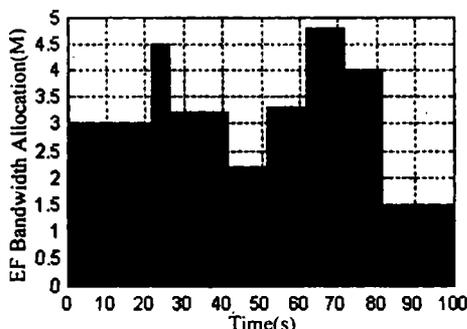


图8 EF类带宽资源0时刻的分配情况

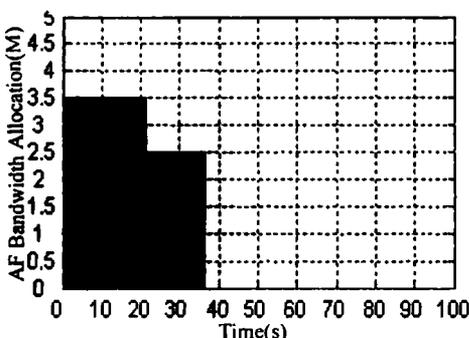


图9 AF类带宽资源0时刻的分配情况

从图8中可以看到,在试验时间内, S_1 和 S_2 对 EF 类带宽资源的最高需求为 4.8M。如果采用静态的带宽分配方式,则整个时间段内带宽分配都必须是 4.8M,对用户而言,将浪费大约

30%的带宽。因此,动态的资源分配方式和用户的利益是相一致的。

结论 要使区分服务模型得到实际的应用,不光要有良好的机制来实现 PHB,也需要动态、灵活的资源分配和管理体制,满足不同用户的 QoS 要求。本文提出了基于 BB 的动态带宽资源分配框架,给出其原型设计。仿真试验表明,该方案能满足动态分配带宽资源的要求,并能实现资源预留。但是,这只是我们工作的第一步。我们将在资源预留下如何接纳控制算法的高效实现、BB 间通信协议的完整设计、BB 的可伸缩性等一系列问题上做进一步的研究。

参考文献

- 1 Iake S, et al. An Architecture for Differentiated Services. RFC 2475, Dec. 1998
- 2 Teitelbaum S, et al. Internet2 Qbone: Building a Testbed for Differentiated Services. IEEE Networks, Sep./Oct. 1999
- 3 Reichmeyer F, et al. A Two-Tier Resource Management Model for Differentiated Services Networks. Internet Draft, draft-rotzy-2-tier-management-00.txt, Nov. 1998
- 4 Nichols K, Jacobson V, Zhang L. A Two-bit Differentiated Services Architecture for the Internet. RFC2638, Apr. 1999
- 5 Fall K, et al. The ns Manual. <http://www.isi.edu/nsnam/ns-documentation>, April 2001
- 6 Pieda P, et al. A Network Simulator. Differentiated Services Implementation. Open IP, Nortel Networks, July 2000
- 7 Jacobson V, Nichols K, Poduri K. An Expedited Forwarding PHB. RFC2598, June 1999
- 8 Heinanen J, Baker F, Weiss W, Wroclawski J. Assured Forwarding PHB Group. RFC2597, June 1999