带冗余策略的分布式 IPSec 网关配置

唐 屹[1],张连宽[2]
TANG Yi[1],ZHANG Lian-kuan[2]

1.广州大学 数学与信息科学学院,广州 510006
2.华南农业大学 数学系,广州 510642
1.Department of Information Sciences,Guangzhou University,Guangzhou 510006,China
2.Department of Mathematics,South China Agricultural University,Guangzhou 510642,China

TANG Yi,ZHANG Lian-kuan.Distributed configuring IPSec gateways with redundant policies.Computer Engineering and Applications,2009,45(3):106-108.

Abstract: An application scenario for IPSec is to partition a network by IPSec gateways.The security requirements are implemented by IPSec policies between gateways.The overlapping tunnels may lead network traffic looping and introduce policy conflicts.It needs policy cuts to avoid those conflicts.However the too fine policies may lead many cryptology computations.In this paper,a distributed gateway configuring method with redundant policy,named DistIPSecR is proposed,to reduce the time-cost computation.We have conducted simulated experiments to validate the proposed method.
Key words: IPSec protocol;security policy;distributed configuring;redundant policy

摘 要:IPSec 协议的一种实现模式是采用 IPSec 网关间隔各个网络段,通过网关的策略配置,满足安全通信需求。然而,策略交叉会导致破坏安全需求的信息回流,拆分策略是避免信息回流的有效方法,但拆分过细,会引发额外的密码计算。提出一种带冗余策略的 IPSec 网关的分布式配置方法,在自动分布式生成无冲突的 IPSec 策略集基础上,引入冗余策略以减少 IPSec 网关的密码计算负荷。模拟实验验证了这种方法的可行性。
关键词:IPSec 协议;安全策略;分布式配置;冗余策略
DOI:10.3778/j.issn.1002-8331.2009.03.031 **文章编号:**1002-8331(2009)03-0106-03 **文献标识码:**A **中图分类号:**TP393

1 引言

IPSec 协议规范了一类基于 IP 层的通用安全服务,可支持 IPv4 和 IPv6 上的各种应用。路由器、防火墙、主机等分布式自主运行的网络节点均可以配置成 IPSec 网关,完成 IPSec 隧道的建立协商,实现端到端的安全通信需求。

在 Intranet 上,IPSec 的一种典型应用模式是采用 IPSec 网关间隔各个网段,依据网段间既定的安全策略,配置适当的 IPSec 网关,实现安全数据传输。一般认为,安全策略可以从需求和实现两个层面来划分[1]。需求确定安全的目标,实现则是为达到安全目标而采取的具体方案。由于需求和实现的不同层面,这使得同一个安全需求可以有不同的实现方案。尽管 IPSec 安全需求的种类有多种,但主要的需求来自安全范围需求[2-3],尤其是在 IPSec 隧道建立以后。

值得注意的是,不适当的 IPSec 实现配置,会引发策略冲突,违背安全需求[1]。一个典型的例子如图 1(a)所示,假设网关 1 与 3 需要完整性检查,而网关 2 与 4 之间需要机密性。当消息到达网关 1 后,1 将信息进行隧道封装等待 3 的检测,信息达到 2 后,2 再将信息加密封装等待 4 的解密。这样信息到达 4 后解密成明文。因为还需要 3 的完整性检测,所以 4 再次发送给 3,3 检测后再发给 4。在最后一次的传递中,3 与 4 传递的是明文,破坏了 3 与 4 之间机密性。

研究表明,当某些网关所涉及的 IPSec 策略发生交叉时,会引发信息回流[1,3],导致解封后的信息流不能满足给定网段的安全需求。解决这个问题的基本方法是在策略交叉处拆分安全策略,以消除信息回流现象。例如图 1(a)所展现的冲突,可以简单地将网关 2 和 4 间的安全策略拆分成两段,实现冲突消解(如图 1(b))。

策略拆分避免了策略冲突,为获得较少的策略数目,通常采用冗余消除的方法来避免过多的等价安全策略。然而,跨段数过少的安全策略集合,可能会使得长距的通信流需要不断进行加封/解封操作,产生不必要的密码计算和安全协商,增加转发网关的无效工作负荷。以图 1 所示的例子,由 1 传至 4 的信息,需要先传至 3,经过解密后,发现其目的地为 4,于是再加密传送给 4。这样,在转发 1~4 的信息时,3 均不可避免地进行加解密的操作。同样的,3 在转发 2~4 的信息时,也需要进行加解密的操作。这使得 3 可能成为数据传输中的一个瓶颈。

考虑引入一条冗余策略 $t_r(1,4)$(如图 1(c))来减少 3 的密码运算量。这种方法尽管增加了策略管理的复杂性,但却使得

基金项目:广东省科技计划(No.2005B10101024)。
作者简介:唐屹(1968-),博士,教授,主要研究:信息安全与人工智能;张连宽,讲师,主要研究:信息安全。
收稿日期:2008-01-04 **修回日期:**2008-04-02

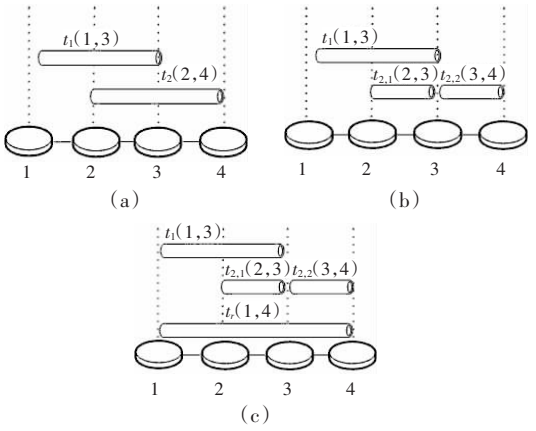


图 1　策略的冲突、消除与冗余

3 在转发 1~4 的信息时,只需判断该网络包是否满足 3、4 间的安全需求,由于 1~4 间的消息传递是需求满足的,这使得 3 不必要进行耗时的密码操作,从而减少了 3 的工作负载。

由于 IPSec 网关自然呈现分布式运行的特性,因此本文将考虑带冗余策略的 IPSec 网关的分布式配置,我们将在已有的基于多智能体(agent)系统的分布式网关配置方法 DistIPSec[4] 的基础上,引入冗余的安全策略,减少 IPSec 网关的密码计算量,提高网关的转发性能。

2　分布式的 IPSec 网关配置

Agent 是描述复杂的分布式问题处理的有效工具[5]。作为一类能够完成预定功能的自主程序,单个 agent 能够依据所获得的局部信息,自主决定程序的执行,而从整体上,多个 agent 可以通过有限的通信,协调完成一个复杂的任务。在基于多 agent 系统的 IPSec 网关配置中,假定每个 IPSec 网关由 agent 进行管理,每个 agent 负责维护与该网关相关的实现策略。

2.1　一些约定

简单起见,记 P 为一个安全策略集合,G 为由 n 个网关构成的集合,$p(\mathrm{src},\mathrm{dst})\in P$ 表示源网关自 src,目的网关为 dst 的安全需求,该需求需要以 ESP 封装的 IPSec 隧道方式来实现。假定对 G 中的 n 个网关进行 1~n 的编号,用 $t(\mathrm{src},\mathrm{dst})$ 表示依据 $p(\mathrm{src},\mathrm{dst})$ 建立的 IPSec 隧道。同时,也采用文献[1]的假设,即网络包在 IPSec 网关间的传送是顺序逐站进行的,例如若网络包采用策略 $t(1,4)$,则意味着该网络包将沿网关1,2,3,4 顺序传送。

记 $START_k=\{t(k,h)|t(k,h)\in P\}$ 表示所有由网关 k 开始的策略,$END_k=\{t(h,k)|t(h,k)\in P\}$ 表示所有由网关 k 结束的策略,而 $PASS_k=\{t(u,v)|t(u,v)\in P$ 且 $u<k<v\}$ 表示所有经过网关 k 的策略,这三个集合构成与网关 k 相关的策略集合 $ASOCTD_k$。

$agent_k$ 管理网关 k 表示它对经过网关 k 的数据流,依据预定的安全策略进行封装/解封操作,同时对 $START_k$ 中的策略进行修改/删除操作。

观察 1 卡氏积 $END_k\times PASS_k$ 上的策略对 $(t(s,k),t(u,v))$,如果满足 $s<u<k<v$,则这两个策略在 k 处冲突,将策略 $t(u,v)$ 拆分为 $t(u,k)$ 与 $t(k,v)$ 来可以消解这个冲突。

观察 2 若存在一组策略 $t(s_{i-1},s_i)$,其中 $s_0=k$,$s_{n-1}=v$,使得 $t(k,v)=t(s_0,s_1)\,\|\,t(s_1,s_2)\,\|\,\cdots\,\|\,t(s_{n-2},s_{n-1})$ 则策略 $t(k,v)$ 是冗余的。这里符号‖表示两个策略首尾相连。

2.2　基本 DistIPSec 思想

DistIPSec 的基本思想是首先依据安全需求,直接产生一个初始的策略集合,然后从这个初始集合出发,寻找等价的策略数较少的无冲突和无冗余的安全策略集。

下文显示了 DistIPSec 的基本过程(更详尽的描述可参见文献[4]),包括:(1)分派安全需求阶段(需求添加 AddPolicy 操作,需求扩散 DiffusePolicy 操作);(2)策略排序阶段(SortRelatedTunnels 操作);(3)冲突检测阶段(冲突检测 CheckConflict 操作,协商通信 SendAndWaitCfltInfo 与 SendAndWaitPrtAllowInfo 操作,分解扩散 PartitionAndDiffusePolicy 操作);(4)冗余检测阶段(检测冗余 CheckConnectingRedundancy 操作,冗余消解 ResolveConnectingRedundancy 操作)。

```
procedure DistIPsec()
    AddPolicy(P,G);                          /* 分派安全需求 */
     for each agent agent_k
       DiffusePolicy(START_k,G);
     endfor;
     repeat
      for each agent agent_k
          SortRelatedTunnels(ASOCTD_k);      /* 策略排序 */
          CheckConflict(END_k,PASS_k);       /* 冲突检测 */
          SendAndWaitCfltInfo(k);
          SendAndWaitPrtAllowInfo(k);
          PartitionAndDiffusePolicy(k,G);
          CheckConnectingRedundancy(END_k,START_k); /* 冗余检测 */
           SendAndWaitCPSInfo(k);
           ResolveConnectingRedundancy();
      endfor;
     until no CfltInfo existed;
endprocedure
```

3　DistIPSecR:带冗余策略的分布式 IPSec 配置

与大多数冲突消解方法一样,DistIPSec 由于寻找策略数较少的策略集,可能会导致长距离的通信在某些网关不断进行额外的密码计算。考虑在 DistIPSec 产生的策略集基础上,构造无冲突的冗余策略,减少额外的密码计算。

3.1　冗余策略构造

在 DistIPSec 产生了无冲突和无冗余的策略集后,$agent_k$ 通过下列计算确定可采用的冗余策略 $t_{r_k}(a,b)$ 的左右边界 a,b:左边界 $a=\min\{a|t(a,k)\in END_k\}$,右边界 $b=\max\{b|t(k,b)\in START_k\}$。显然,策略 $t_{r_k}(a,b)$ 是非冲突的策略,因为若 $t_{r_k}(a,b)$ 冲突,则存在策略 $t(c,d)$ 使得 $a<c<b<d$。若 $t(c,d)\in PASS_k$,则网关 k 为冲突点,策略 $t(a,k)$ 与 $t(c,d)$ 也冲突;而当 $c<d<k$,则 $t(a,k)$ 与 $t(c,d)$ 冲突,$k<c<d$,则 $t(k,b)$ 与 $t(c,d)$ 冲突。这些引发冲突的策略来自 DistIPSec 直接产生的策略集,这与已有的结果矛盾,因此,策略 $t_{r_k}(a,b)$ 是非冲突的策略。

注意到,如果引入冗余策略 $t_{r_k}(a,b)$,则网关 k 至少可以减少由 a 经 k 转发至 b 的网络数据的密码计算。

3.2　冗余策略确定

引入冗余策略,其目的是减少密码运算量,而每个网关的密码运算负载减少量,依赖于经过这个网关的网络流量。设评估函数 $eval(k)$ 反映引入冗余策略 $t_{r_k}(a,b)$ 后,网关 k 可以减少

的由 a 经 k 转发至 b 的密码计算量。于是,可以利用基于多 agent 的问题求解的非完全方法[6],寻求密码计算减少量的近似最佳解,基本步骤如下:

步骤 1 $agent_k$ 向 $agent_a, agent_{a+1}, \cdots, agent_{k-1}, agent_{k+1}, \cdots, agent_b$ 发送带减少量 $eval(k)$ 的消息 *improve*;

步骤 2 $agent_k$ 比较 $eval(k)$ 和接收到的减少量,若发现 $eval(k)$ 最大,则增加一条冗余策略,向相关的 *agent* 发送 ok 消息;若发现非最大,则保持原来的策略数,发送 keep 消息;在比较过程中,若存在评估值相同的消息,则约定这些消息中,编号最小的网关是最大减少量来源。

步骤 3 若 $agent_k$ 没有收到 ok 消息或接收的 ok 消息涉及的策略不在集合 $PASS_k$ 中,则重复步骤 1 和步骤 2,直到整个系统没有 ok 消息出现。

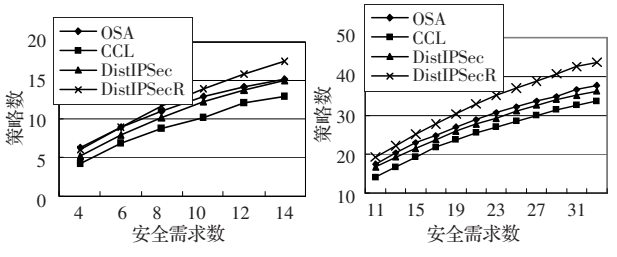
3.3 评估函数的一种确定方法

如何确定评估函数估计可减少的密码计算次数并不是件容易的事,这涉及实际的网关间的流量。作为一个可能的评估函数,假定网关间的流量依赖于所谓的局部性原则,即网关编号相近的两个网关间的流量要比网关编号相隔较远的两个网关间的流量要大,这使得评估值可以简化为安全策略中的源地址和目的地址的差值。

4 实验与讨论

进行了计算机模拟实验,验证 DistIPSecR 方法。首先随机生成给定数量的安全需求集,然后基于 DistIPSec,生成初始的无冲突的安全集,在这基础上,应用 DistIPSecR,生成带冗余策略的策略集,所采用的评估函数基于流量局部性原则。为评价减少的密码计算次数,假定两个网关节点 i,j 以概率 $t\cdot\alpha^{-|i-j|}$ 产生通信包,其中 t 为规范化因子,α 为参数,取值范围为 1.1~2.5,为与已有的数据相比较,分别模拟了 20 个网关(6 组安全需求数量)和 50 个网关(12 组安全需求数量)的情形。

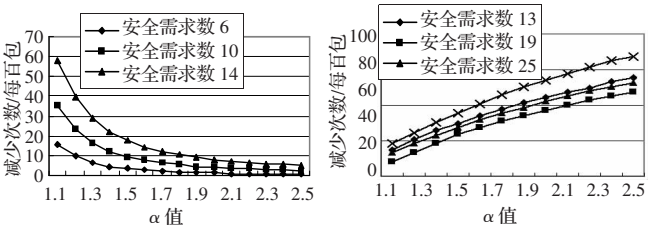
在每种情形下,分别生成 1 000 个给定安全需求数量的安全需求集,针对所产生的安全需求,统计 DistIPSecR 方法产生的无冲突策略数,并与 DistIPSec,OSA[2]和 CCL[3](近似数据来自文献[3])相比较,比较结果如图 2 所示。



(a)20-网关 (b)50-网关

图 2 DistIPSecR、DistIPSec、OSA、CCL **间的比较**

为反映密码计算的减少程度,首先随机选取一个网关,然后依据上述的概率产生通信包,如果这个通信包刚好需要使用一个冗余策略,则表明这个通信包可以减少一次密码计算,在给定安全需求数量的情况下,按照上述的方法生成 10 000 个网络包,统计其中每百个包的平均密码计算减少次数,结果如图 3 所示。



(a)20-网关 (b)50-网关

图 3 DistIPSecR **的密码计算减少次数**

依据实验结果,DistIPSecR 方法是这四种方法中,所需安全策略数最多的方法,在 20-网关情形,平均比 DistIPSec 方法多生成 14.5%的策略,而在 50-网关情形,则平均多生成 18.1%的策略,这些多的策略就是新增加的冗余策略。

新增加的这些冗余策略带来了密码计算量的减少。图 3 分别显示了模拟的每百个网络包所带来的密码计算减少次数,发现,随着安全需求数的增加,密码计算减少次数也增加,这显示,当一个网络的安全需求数较多时,DistIPSecR 可能带来较多的密码计算次数。另一方面,依据本文的算法,当各个网关作为通信发起方的概率相同时,α 值(alpha 值)越大,反映通信的局部化程度越高,少量增加的冗余策略由于其分布的稀疏,在较大的 α 取值下,不会大幅度减少密码计算次数,而当通信的局部化程度较低时,会带来比较理想的减少量。

5 结束语

IPSec 网关的分布式配置方法,有助于增加管理和配置无冲突 IPSec 策略的灵活性。为避免安全策略发生冲突,拆分策略是简单有效的方法,但过细的策略配置会引发额外的密码计算,增加网关的工作负荷。讨论一种带冗余策略的 IPSec 网关的分布式配置方法 DistIPSecR,在自动分布式生成无冲突的 IPSec 策略集基础上,引入冗余策略,减少 IPSec 网关的密码计算负荷。所进行的模拟实验验证了这种方法的可行性。

参考文献:

[1] Fu Z,Wu S,Huang H,et al.IPsec/VPN security policy: correctness,conflict detection and resolution[C]//IEEE Policy 2001 Workshop,2001:39-56.

[2] Yang Y,Martel C,Wu S.On building the minimal number of tunnels-an ordered-split approach to manage IPSec/VPN policies[C]//Proceedings of NOMS'04,2004:277-290.

[3] Chang C,Chiu Y,Lei C.Automatic generation to conflict-free IPSec policies[C]//Proceedings of FTNDS'05,2005:233-246.

[4] 唐屹,张连宽.IPSec 网关的一种分布式配置方法[J].计算机工程与应用,2008,44(14):127-129.

[5] Liu J,Jing H,Tang Y.Multi-agent oriented constraint satisfaction[J].Artificial Intelligence,2002,136(1):101-144.

[6] Hirayama K,Yokoo M.The distributed breakout algorithm[J].Artificial Intelligence,2005,161(1/2):89-116.