新手入门：网络基础

协议解析

新手入门：网络基础协议解析

就象我们说话用某种语言一样，在网络上的各台计算机之间也有一种语言，这就是网络协议，不同的计算机之间必须使用相同的网络协议才能进行通信。

网络协议是网络中互相通信的对等实体间，交换信息所必须遵守的规则。当前计算机网络的体系结构是以 TCP/IP 协议为主的 Internet 结构。

本技术手册将为你详细介绍 TCP/IP 中最重要、最常用的协议。

TCP/IP 协议总述

TCP/IP 是互联网中的基本通信语言或协议。它是以一个两层的程序。高层为传输控制协议（TCP），负责将信息或文件装配到更小的包中。低层是网际协议（IP），处理每个包的地址部分。

- 网络协议指南
- 网络一词——TCP/IP

基本协议定义和工作原理

位于网络层的 IP 协议可能是网络通信中最重要也是最著名的协议，它使我们能够唯一标识 Internet 上的每台电脑；RIP 协议帮助我们寻找最近的路由器。位于传输层的 TCP、UDP 协议是解决计算机程序之间的通信问题两种的方法，本节将主要介绍这四个协议。

- IP 协议介绍
- IP 协议报头
- 什么是路由信息协议 RIP
- 用户数据报协议 UDP
- API 公共数据选择：套接字和 UDP
- TCP 协议的通讯方式

安全协议概念及实现方法

光有上述保证网络工作的基本协议就够了么？面对互联网汹涌的安全攻击，我们还需要保证信息安全的协议：IPSec 和 SSL。IPSec 通过加密的安全服务确保在 IP 网络上进行保密而安全的通讯；SSL(安全套接层)是一个加密协议，提供加密和身份识别服务。

- IPsec：IP 层协议安全结构
- 实现 VPN 的 IPsec 协议细节
- 详解 SSL 协议
- 黑帽大会 2010：SSL 协议使用的最新测试细节

网络协议指南

在网络学和通信学中，协议就是定义过程的正式规格说明书，当传送或者接受数据的时候必须严格遵守。协议定义了网络中数据传输的格式、时间选择、先后次序、错误检查。

简单的说，上述说法意味着如果你想让两台或者两台以上的设备进行通信，它们就需要一个共同的协议或者说是一组规则来指导这些设备在什么时候、以什么方式进行相互之间会话。

网络协议一般由 RFC（requests for comments，请求注释，Internet 标准草案定义组织）来定义，该组织中的 IETF（Internet 工程任务组，Internet Engineering Task Group）负责制定新的标准的或者协议。网络产品供应商（例如 IBM、思科、微软、Novell）就根据这些标准并将这些标准在他们的产品中实现出来。

已经出台的协议有成百上千个，将他们一一列举在这里是不可能的。所以我们现在仅仅介绍一下几个最常用的协议，而且，我们将在未来的文章中研究一些专业性更强的协议。

需要紧记在心的事情是，当你从最下边的物理层到最上边的应用层的时候，设备用于处理协议的时间将越来越长。

（来源：TechTarget 中国　作者：Firewall.cx　译者：王菲）

网络一词——TCP/IP

TCP/IP（传输控制协议/网际协议，Transmission Control Protocol/Internet Protocol）是互联网中的基本通信语言或协议。在专用网络（不管是内联网还是外联网）中，它也被用作通信协议。当你直接连接网络时，你的计算机就会有一个 TCP/IP 程序的副本，此时接收你所发送的信息的计算机也应有一个 TCP/IP 程序的副本。

TCP/IP 是一个两层的程序。高层为传输控制协议（TCP），它负责将信息或文件装配到更小的包中。这些包通过网络传送到接收端的 TCP 层，接收端的 TCP 层把包还原为原始文件。低层是网际协议（IP），它处理每个包的地址部分，使这些包正确的到达目的地。网络上的网关计算机根据信息的地址来进行路由选择。即使来自同一文件的分包路由也有可能不同，但它们最后会在目的地汇合。

TCP/IP 使用客户机与服务器模式进行通信。TCP/IP 通信是点对点的，意思是通信是网络中的一台主机与另一台主机之间的。TCP/IP 与上层应用程序之间可以说是“无状态的”，因为每个客户请求都被看做是与上一个请求无关的新请求。正是它们之间的“无状态”释放了网络路径，才使每个人都可以连续不断的使用网络。（请注意，TCP 层本身并不是无状态的。）

许多用户熟悉使用 TCP/IP 协议的高层应用协议。包括万维网的超文本传输协议（HTTP）、文件传输协议（FTP）、Telnet（它可以让你登录到远程计算机）和简单邮件传输协议（SMTP）。这些协议通常和 TCP/IP 协议打包在一起。

使用模拟电话调制解调器连接网络的个人电脑通常使用串行线路 IP 协议（SLIP）和点对点协议（PPP）。这些协议封装在 IP 包中，这样它们就可以通过拨号连接发送到接入供应商的调制解调器中。

与 TCP/IP 协议相关的协议还包括用户数据报协议（UDP），它代替 TCP/IP 协议来达到特殊的目的。网络主机用来交换路由信息的其他协议有 Internet 控制信息协议（ICMP）、内部网关协议（IGP）、外部网关协议（EGP）和边界网关协议（BGP）。

（来源：TechTarget 中国）

IP 协议介绍

Internet 协议（或者叫 IP 协议）可能是网络通信中最重要也是最著名的协议之一，它使我们能够唯一标识网络中（这里一般指企业内部的网络）或者 Internet 上的每一台电脑。

当将一台计算机连入网络中或者连入 Internet 中的时候，它将被分配一个唯一的 IP 地址。如果你是将它连入 Internet 中，IP 地址的分配是由你所在的 ISP（网络服务提供商）自动完成的，如果你是将其连入到一个局域网（LAN）中，那么你的 IP 地址可以是自动分配的，你也可以按照分配给你的 IP 地址，在你的工作站上进行手动配置。

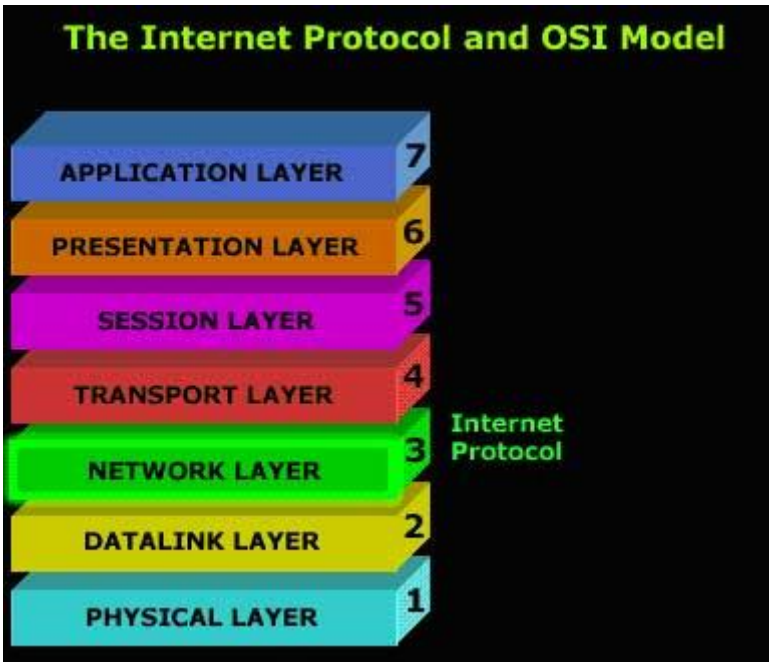
如果你想真正的了解网络通信是如何工作的，那么深入了解 IP 协议就是不得不强调的事情，DNS（域名服务器）、FTP（文件传输协议）、SNMP（简单网络管理协议）、HTTP（超文本传输协议）及其很多其他的协议和服务都需要依赖 IP 协议才能正常发挥功能，所以你立刻就能看到 IP 协议将不仅仅是你工作站上的一个地址而已！

现在，因为 IP 协议是一个含有丰富知识的学科，我们不可能在一两页的文章中涵盖其全部内容，所以决定将其分成几个不同的部分，以便于使其更易懂易学。

（来源：TechTarget 中国　作者：Firewall.cx）

IP 协议报头

就象所有其他的协议一样，IP 协议在 OSI 模型占有一席之地，因为它是一个如此重要的网络以至于其他协议都依赖于它，所以 IP 协议需要先于其他协议放入 OSI 模型中，这就是你会在 OSI 模型的第三层发现它的原因（其它的应用协议基本上都在三层以上）。



当一个计算机接受到来自网络的数据包的时候，它将首先在数据链路层（第二层）检查数据包中包含的目标机 MAC 地址，如果 MAC 地址于本机匹配，它才会将数据包传递给网络层。

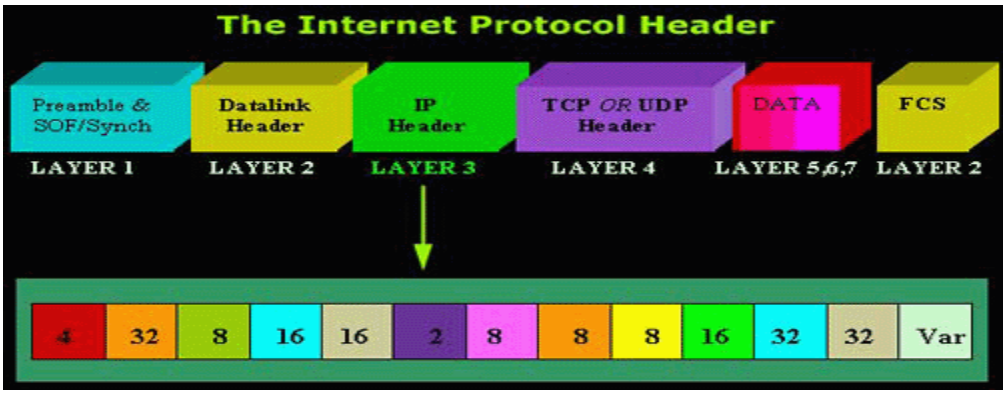
在网络层，计算机将检查数据包中的目标 IP 部分是否与本机的 IP 地址匹配（如果该数据包是一个广播数据包，则会无条件的通过网络层）。

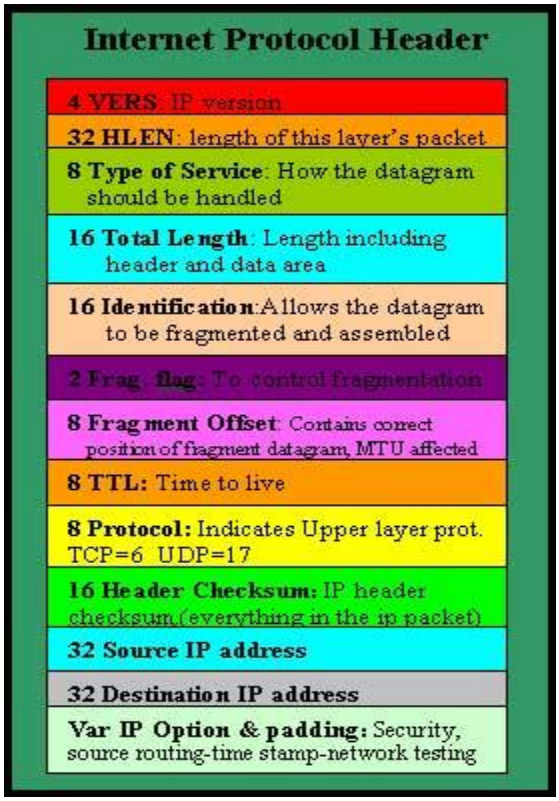
从那儿开始，数据包就被上层的按照要求进行处理。

另一方面，计算机也可能产生一个数据包并将其送入网络，然后，当这个数据包沿着 OSI 模型向下传送到达网络层的时候，目标机的源主机（也就是本主机）的 IP 地址就被加入到 IP 头中去了。

IP 报头

现在我们开始来分析 IP 协议报文头，你可以看到它都有分成那些值域以及这些值域的位置安排，在 IP 头里边你可以找到对于每个使用该协议的数据包都至关重要的目标机和源主机 IP 地址。





值得一提的是第九个值域——“协议”值域，它包含一些重要的信息，一旦计算机将 IP 剥离，它将告诉该计算机将该数据包送到那里。如果你还记得，OSI 模型中的第四层（我们叫它传输层）中存在着 TCP 协议和 UDP 协议，当数据包到达一个计算机并且被网络层下边各层进

行了处理以后，就需要知道将该数据包送往上层的什么地方。这个值域就是告诉计算机将剩下的数据是送给传输层的 TCP 协议还是送给 UDP 协议。

目标机 IP 地址是另一个非常重要的值域，该值域包含目标主机的 IP 地址。

（来源：TechTarget 中国）

什么是路由信息协议 RIP

路由信息协议（RIP）

路由信息协议原来是为施乐 PUP（PARC 通用协议）设计的。路由信息协议在 1981 年的施乐网络系统协议集中称做“GWINFO”，并且在 1988 年定义为“RFC 1058”。路由信息协议很容易配置，在小型网络中工作得非常好。然而，在大型网络中，路由信息协议的工作效率不高。正如我对自己说的那样，路由信息协议会使你的网络四分五裂。在大型网络环境中，有替代路由信息协议的协议。

路由信息协议的特点如下：

- 开放式协议，广泛应用，稳定。
- 适用于小型网络，很容易配置。
- 有适用于 Novell 和 Appletalk 软件的类似于路由信息协议的距离向量路由协议。
- 内部网关协议（IGP）
- IP 路由信息协议更新每 30 秒通过广播发送一次（所有 RIPv2 路由器多播地址是 224.0.0.9）。
- UDP（用户数据报协议）端口 520。
- 可管理距离为 120。
- 单一衡量标准是计算跳数（极限是 15，计数到无穷大）
- 计时器帮助调节性能：

——更新计时器--路由更新的频率。

——每 30 秒钟 IP 路由信息协议发送一次完整的路由表,依据水平分裂情况而定。（IPX 路由信息协议每隔 60 秒发送一次路由表）。

——非法计时器--在路由更新中没有刷新的内容。路由信息协议等待 180 秒，然后把一个路由标记为非法并且立即让这个路由处于抑制计时状态。

——保持计时和触发更新--在思科幻境中以稳定的路由提供帮助。抑制计时可确保正常的更新信息不适当地引起路由循环。路由器在特定的时间段内不对非优越的信息作出反应。路由信息协议的抑制计时时间是 180 秒。

——刷新计时器--路由信息协议在把路由确实从路由表中删除之前，还要额外等待 240 秒。

- 其它帮助路由循环的稳定功能还包括：

——水平分裂

——毒性逆转

- Bellman-Ford 算法
- RIPv2 支持 VLSM 和自动摘要。

（来源：TechTarget 中国　作者：Donna Harrington 译者：王菲）

用户数据报协议 UDP

用户数据报协议(UDP，User Datagram Protocol)是当消息在使用 IP 协议的网络中的计算机之间交换时通过有限服务的传输协议。用户数据报协议是传输控制协议（TCP）的替代协议，与 IP 协议一起，有时候也被称为 UDP/IP 。与传输控制协议类似的是，用户数据报协议使用了 IP 协议来实际获得从一台计算机发送到另一台计算机上的数据单元（称为数据报）。然而，传输控制协议不同的是，用户数据报协议不提供将消息分割成包的（数据报），并在另一端重新组装的功能。特别的是，用户数据报协议不提供数据到达的包的顺序。这就意味着使用用户数据报协议的应用程序必须能够确保全部的消息都能够到达，并是按正确的顺序。那些交换非常小的数据单元（因此也只有非常少的消息需要重新组装）以便于节省处理时间的网络应用程序更倾向于使用用户数据报协议，而不是传输控制协议。一般的文件传输协议（TFTP）使用用户数据报协议来替代传输控制协议。

用户数据报协议提供了 IP 层没有提供的两个服务。它提供了端口号来帮助辨别不同的用户请求，以及（可选的）验证数据完整达到，未经修改的检测功能。

在开发系统互连（OSI）通信模型中，用户数据报协议，与传输控制协议一样，都是在第四层，传输层。　　总体来说，这 5 个 IPv6 规划难题不应该成为任何人不敢尝试 IPv6 迁移的原因，它们应该起到提醒的作用，即实施 IPv6 需要大量的提前规划。我们很多客户已经受益于 IPv6 专业人员公司提供的专业服务。然而实际的实施还是很少的，我们期望第三方服务将随着实际的部署而增加。重视这些 IPv6 规划难题，通往 IPv6 之路将变得更加清晰。

（来源：TechTarget 中国）

API 公共数据选择：套接字和 UDP

开发者刚接触 Web 服务的世界时，往往认为只有 HTTP 条款。有很多方法可以从服务端传输你的数据到客户端，但是在列出这些例子之前，让我们先谈谈“OSI 模型的通信系统”。人们思考系统间通信的习惯已经大大影响了开放系统互连(OSI)模型，它规定了从物理硬件(1)到最高的应用层的七“层”系统概念。在本文中，我们主要关注传输（4）层和更高层。首先，让我们看一下用单独的客户端连接服务器的方法。欲了解更多信息，请看我的文章“数据传输格式”。

套接字

Socket 是一种程序语言，用来表示进程间连接的抽象概念，通常是通过计算机网络。这个概念出现在 80 年代早期，使用 C 库的 BSD Unix 操作系统中，被证明非常有用，在大多数语言中，soacket API 是计算机间通信最基础的技术。网络 socket 使用 IP（因特网协议）地址和“端口”号来指定计算机和通信流程的。

有很多关于端口号使用的协议，但是我不打算在这里引入。实际上的低级 socket 接口的概念和物理硬件是从事于操作系统的。TCP 和 UDP 接口协议建立在 socket 之上的，但是你也能够在“原始套接字”层上编程。原始的 soceket 通信非常快，但是编程非常复杂、不便携。在 Java 中学习更多的 socket 编程。

优点：

- 最快的通信速度
- 灵活的数据量

缺点：

- 需要相当多的自定义编程
- 在不同的操作系统之间，可移植性有限

用户数据报协议（UDP）

最简单的因特尔协议在通信进程间不能提供自动错误检查或者“握手”检验，因此，程序员有时会调用一些不可靠的数据报协议。UDP 数据包可以解决一个或者多个接收者的问题。如果丢失一个或者多个数据包，不能毁坏客户的应用程序，你只能够使用 UDP。例子包括流媒体和游戏。

优点：

- 快速
- 一个或者多个接收者

缺点：

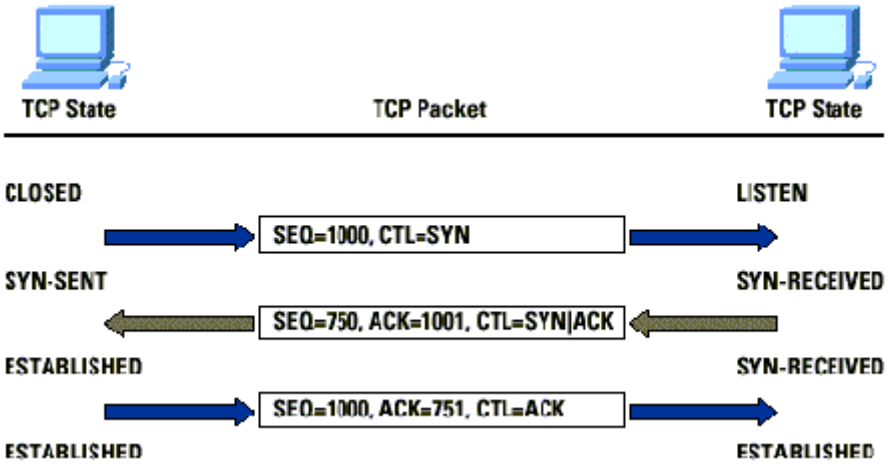
- 只能在数据丢失并可以容错时使用
- 不是新的，但它还未广泛应用，所以可能会出现我们未曾遇到过的奇怪问题。

(来源：TechTarget 中国　作者：William Brogden　译者：刘志超)

TCP 协议的通讯方式

一、TCP 三次握手

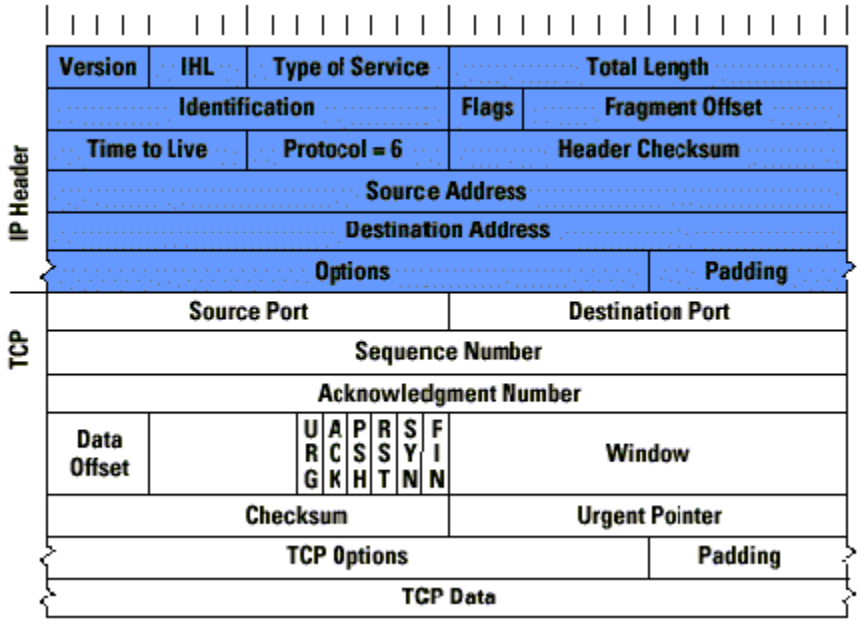
传输控制协议(Transport Control Protocol)是一种面向连接的，可靠的传输层协议。面向连接是指一次正常的 TCP 传输需要通过在 TCP 客户端和 TCP 服务端建立特定的虚电路连接来完成，该过程通常被称为“三次握手”。可靠性可以通过很多种方法来提供保证，在这里我们关心的是数据序列和确认。TCP 通过数据分段(Segment)中的序列号保证所有传输的数据可以在远端按照正常的次序进行重组，而且通过确认保证数据传输的完整性。要通过 TCP 传输数据，必须在两端主机之间建立连接。举例说明，TCP 客户端需要和 TCP 服务端建立连接，过程如下所示：



在第一步中，客户端向服务端提出连接请求。这时 TCP SYN 标志置位。客户端告诉服务端序列号区域合法，需要检查。客户端在 TCP 报头的序列号区中插入自己的 ISN。服务端收到该 TCP 分段后，在第二步以自己的 ISN 回应(SYN 标志置位)，同时确认收到客户端的第一个 TCP 分段(ACK 标志置位)。在第三步中，客户端确认收到服务端的 ISN(ACK 标志置位)。到此为止建立完整的 TCP 连接，开始全双工模式的数据传输过程。

二、TCP 标志

这里有必要介绍一下 TCP 分段中的标志(Flag)置位情况。如下图所示。



*SYN：同步标志

同步序列编号(Synchronize Sequence Numbers)栏有效。该标志仅在三次握手建立 TCP 连接时有效。它提示 TCP 连接的服务端检查序列编号，该序列编号为 TCP 连接初始端(一般是客户端)的初始序列编号。在这里，可以把 TCP 序列编号看作是一个范围从 0 到 4，294，967，295 的 32 位计数器。通过 TCP 连接交换的数据中每一个字节都经过序列编号。在 TCP 报头中的序列编号栏包括了 TCP 分段中第一个字节的序列编号。

*ACK：确认标志

确认编号(Acknowledgement Number)栏有效。大多数情况下该标志位是置位的。TCP 报头内的确认编号栏内包含的确认编号(w+1，Figure-1)为下一个预期的序列编号，同时提示远端系统已经成功接收所有数据。

*RST：复位标志

复位标志有效。用于复位相应的 TCP 连接。

*URG：紧急标志

紧急(The urgent pointer) 标志有效。紧急标志置位，

*PSH：推标志

该标志置位时，接收端不将该数据进行队列处理，而是尽可能快将数据转由应用处理。在处理 telnet 或 rlogin 等交互模式的连接时，该标志总是置位的。

*FIN：结束标志

带有该标志置位的数据包用来结束一个 TCP 回话，但对应端口仍处于开放状态，准备接收后续数据。

三、TCP 端口

为了能够支持同时发生的并行访问请求，TCP 提供一种叫做“端口”的用户接口。端口是操作系统核心用来识别不同的网络回话过程。这是一个严格的传输层定义。通过 TCP 端口和 IP 地址的配合使用，可以提供到达终端的通讯手段。实际上，在任一时刻的互联网络连接可以由 4 个数字进行描述： 来源 IP 地址和来源端口，目的 IP 地址和目的端口。位于不同系统平台，用来提供服务的一端通过标准的端口提供相应服务。举例来说，标准的 TELNET 守护进程(telnet daemon)通过监听 TCP 23 端口，准备接收用户端的连接请求。

四、TCP 缓存(TCP Backlog)

通常情况下，操作系统会使用一块限定的内存来处理 TCP 连接请求。每当用户端发送的 SYN 标志置位连接请求到服务端的一个合法端口(提供 TCP 服务的一端监听该端口)时，处理所有连接请求的内存使用量必须进行限定。如果不进行限定，系统会因处理大量的 TCP 连接请求而耗尽内存，这在某种程度上可以说是一种简单的 DoS 攻击。这块经过限定的，用于处理 TCP 连接的内存称为 TCP 缓存(TCP Backlog)，它实际上是用于处理进站(inbound)连接请求的一个队列。该队列保存那些处于半开放(half-open)状态的 TCP 连接项目，和已建立完整连接但仍未由应用程序通过 accept()调用提取的项目。如果这个缓存队列被填满，除非可以及时处理队列中的项目，否则任何其它新的 TCP 连接请求会被丢弃。

一般情况下，该缓存队列的容量很小。原因很简单，在正常的情况下 TCP 可以很好的处理连接请求。如果当缓存队列填满的时候新的客户端连接请求被丢弃，客户端只需要简单的重新发送连接请求，服务端有时间清空缓存队列以相应新的连接请求。

在现实环境中，不同操作系统支持 TCP 缓冲队列有所不同。在 BSD 结构的系统中，如下所示：

OS	Backlog	BL+ Grace	Notes
SunOS 4.x.x	5	8	
IRIX 5.2	5	8	
Linux 1.2.x	10	10	Linux does not have this grace margin
FreeBSD 2.1.0		32	
FreeBSD 2.1.5		128	
Win NTs 3.5.1	6	6	NT does not appear to have this margin
Win NTw 4.0	6	6	NT has a pathetic backlog

五、TCP 进站(Inbound)处理过程

为了更好的讲述 TCP SYN Flood 的攻击过程，我们先来介绍一下正常情况下，TCP 进站处理的过程。

服务端处于监听状态，客户端用于建立连接请求的数据包(IP packet)按照 TCP/IP 协议堆栈组合成为 TCP 处理的分段(segment)。

分析报头信息：TCP 层接收到相应的 TCP 和 IP 报头，将这些信息存储到内存中。

检查 TCP 校验和(checksum)：标准的校验和位于分段之中(Figure-2)。如果检验失败，不返回确认，该分段丢弃，并等待客户端进行重传。

查找协议控制块(PCB{})：TCP 查找与该连接相关联的协议控制块。如果没有找到，TCP 将该分段丢弃并返回 RST。(这就是 TCP 处理没有端口监听情况下的机制) 如果该协议控制块存

在，但状态为关闭，服务端不调用 connect()或 listen()。该分段丢弃，但不返回 RST。客户端会尝试重新建立连接请求。

建立新的 socket：当处于监听状态的 socket 收到该分段时，会建立一个子 socket，同时还有 socket{}，tcpcb{}和 pcb{}建立。这时如果有错误发生，会通过标志位来拆除相应的 socket 和释放内存，TCP 连接失败。如果缓存队列处于填满状态，TCP 认为有错误发生，所有的后续连接请求会被拒绝。这里可以看出 SYN Flood 攻击是如何起作用的。

丢弃：如果该分段中的标志为 RST 或 ACK，或者没有 SYN 标志，则该分段丢弃。并释放相应的内存。

(来源：TechTarget 中国)

IPsec：IP层协议安全结构

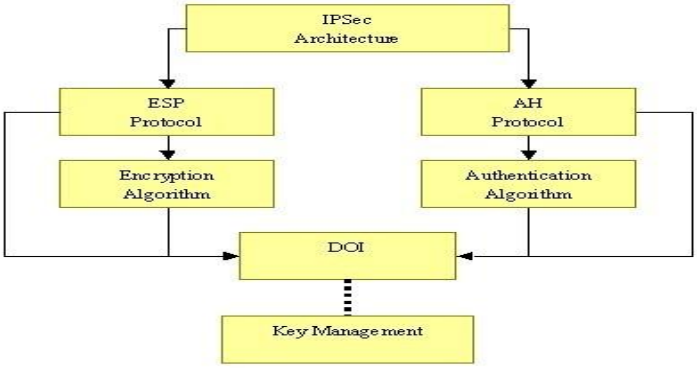
IPsec 在 IP 层提供安全服务，它使系统能按需选择安全协议，决定服务所使用的算法及放置需求服务所需密钥到相应位置。 IPsec 用来保护一条或多条主机与主机间、安全网关与安全网关间、安全网关与主机间的路径。

IPsec 能提供的安全服务集包括访问控制、无连接的完整性、数据源认证、拒绝重发包（部分序列完整性形式）、保密性和有限传输流保密性。因为这些服务均在 IP 层提供，所以任何高层协议均能使用它们，例如 TCP 、 UDP 、ICMP 、 BGP 等等。

这些目标是通过使用两大传输安全协议，头部认证（AH） 和封装安全负载 （ESP），以及密钥管理程序和协议的使用来完成的。所需的 IPsec 协议集内容及其使用的方式是由用户、应用程序、和/或站点、组织对安全和系统的需求来决定。

当正确的实现、使用这些机制时，它们不应该对不使用这些安全机制保护传输的用户、主机和其他英特网部分产生负面的影响。这些机制也被设计成算法独立的。这种模块性允许选择不同的算法集而不影响其他部分的实现。例如：如果需要，不同的用户通讯可以采用不同的算法集。

定义一个标准的默认算法集可以使得全球因英特网更容易协同工作。这些算法辅以 IPsec 传输保护和密钥管理协议的使用为系统和应用开发者部署高质量的因特网层的加密的安全技术提供了途径。IPsec 结构包括众多协议和算法。这些协议之间的相互关系如图所示。



IP 层协议安全结构

（来源：TechTarget 中国）

实现 VPN 的 IPsec 协议细节

Cisco 的 IPsec（Internet Protocol Security）套件的 IOS 实现是一个基于开放标准的框架，它提供给管理员各种安全 IP 网络通信的工具。

IPsec 框架是一套 IETF 标准，它们用于非安全网络中数据安全传输，比如 Internet。IPsec 提供了网络层的安全通信协议，以及交换身份验证和安全性协议管理信息的机制。IPsec 套件是用于解决 IPv4 的一些基本的安全缺陷。IPv4 是用于一个数据包交换网络环境中运行有或没有操作系统的计算机之间共享信息。可以这样说，在 70 年代 TCP/IP 开发出来时，安全性的重要性还比不上数据包准确传输。然而，随着全球网络的大量系统数以百万计计算机之间的 TCP/IP 数据交换支持的增长，这些对安全性的攻击成为了关注点。

IPsec 对抗安全性攻击

目前有三种主要的 IP 安全性攻击。其中有两个是针对于有全局唯一 IP 地址的互相独立的主机间的 IP 数据传输。

IP 欺骗。为了保证收到的信息是可信的，信息的来源也必须是可信的。IP 数据包发送是逐跳式（Hop-by-Hop）处理的。如果一个攻击者知道网络拓扑结构，他就能够让一个系统失效和伪装或“欺骗”它的身份。因为大多数 IP 安全规范都是围绕主机的 IP 地址的，IP 欺骗对于需要安全地交换数据的网络管理员是很大的问题。

会话劫持（Session hijacking）是比 IP 欺骗更高级别的攻击。这时攻击者会失效所欺骗的主机并伪装活跃的网络会话。这是一个比伪装主机 IP 地址更狡猾的攻击，因为它的成功运行也依赖于软件攻击。

流量嗅探（Traffic Sniffing）是包交换网络所特有的，在包交换网络中所有的数据包对于所有连接到传输介质的网络节点都是可见的。不管是什么样的传输介质，路由器、交换机和防火墙对通过这些网关的数据包都有可见权。而这使得这些设备可以很好地应付被支过滤 IP 流量，同时这使它们成为收集 IP 数据包并提取数据内容的最佳位置。

IPsec 协议细节

为了解决这些攻击，IETF 开发了一些不同的协议标准定义。这些标准提供了四个基本的服务：

1. 数据传输加密：原始主机能够在数据包传输前对其进行加密；

2. 数据完整性验证：接收主机能够对每一个用来保证原始数据传输被接收而发送的数据包进行验证；

3. 数据来源验证：原始主机能够标记数据包，使得接收者能够对它们进行验证；

4. 数据状态完整性：原始和接收主机都能够标记数据包，所以任何数据流的重传输都可以被检测和拒绝（也就是 Anti-replay）；

IPsec 使用许多不同的安全性协议来支持它们的服务。从一般层次上，这些协议可以分成两个不同分组：数据包协议和服务协议。数据包协议用以提供数据安全性服务。其中有两种 IPsec 数据包协议：认证报头（Authentication Header，AH）和封装安全载荷（Encapsulating Security Payload，ESP）。而服务协议有很多种，但其中主要的一种是 Internet 密钥交换协议（Internet Key Exchange Protocol，IKE）。下面是 IOS IPsec 实现的协议概况：

认证报头（Authentication Header）：AH 是定义在 IETF RFC 2402 的，它支持 IPsec 数据验证、认证和完整性服务。它不支持数据加密。典型情况下 AH 是单独实现的，但它也可以与 ESP 一起实现。在仅仅需要保证数据交换安全的时候，我们才使用 AH。既然 AH 不支持数据加密，你也许会问为什么我们还要用 AH？你可以这样来看这个问题：如果应用已经支持数据加密，那就不需要额外的数据加密。相对于 ESP 而言，AH 在处理开销上是更“轻”的，所以它更容易应用在低端的路由器上。

此外，相对于 ESP，AH 提供了更好的 IP 层安全性。AH 通过为所有在传输中不被修改的 IP 数据包信息生成 Hash 签名数据来保证 IP 数据包的安全。AH 安全性数据是存储在 32 位长度的 AH 报头中的，而这是安装在 IP 数据报头和 4 层协议报头之间的。因为 AH 是负责使 IP 数据包“安全”的，所以 AH 不能部署在使用 Network Address Translation (NAT)的网络环境中。AH 是在传输模式或通道模式中的起作用的。大多数情况下我们会使用通道模式，而将原始 IP 数据包封装在一个新的 AH 安全 IP 数据包中。这个新的数据包包含一个新的 IP 报头（其中有 IPsec 远程节点网关的目标地址）和 AH 报头，接着是原始 IP 数据包和四层报文。IANA（Internet Assigned Numbers Authority）将 ESP 的协议 ID 赋为 51。

封装安全载荷（Encapsulating Security Payload）：ESP 是定义在 IETF REF 2406 中的，它支持 IPsec 数据加密、验证、认证和和完整性服务。ESP 可以单独实现或者与 AH 一起实现。AH 报头是预先就包含在 IP 数据包的数据负载部分的，而 ESP 将 IP 数据包中的整个数据部分和一个报头及报尾封装在一起。ESP 报头包含安全性和序列化信息。ESP 报尾包含补充参数和（必要的）验证数据。ESP 对原始 ULP 数据及其密文的封装要求比 AH 更多的路由器处理资源。此外，ESP 也要求将 1500-byte Layer 4 报文进行分片，以支持额外的安全负载数

据。类似于 AH，ESP 也支持传输和通道操作地方，但几乎所有供应商都专门实现了通道模式。ESP RFC 没有规定哪个协议必须用于加密数据。Cisco IOS 支持 56-DES、3DES 和 AES 的加密协议。还有其它一些供应商也实现了 Blowfish 和 IDEA。IANA 将 ESP 的协议 ID 赋为 50。

Internet 安全联系和密钥管理协议（Internet Security Association and Key Management Protocol，ISAKMP）和 Internet 密钥交换（Internet Key Exchange，IKE）：这些协议提供了实现 IPsec VPN 服务协商的框架和过程。ISAKMP 是定义在 IETF REF 2408 中，而 IKE 定义在 IETF RFC 2409 中。ISAKMP 定义了创建和删除认证密钥和安全联系（SA）的计划、语法和程序。IPsec 节点使用 SA 去跟踪不同 IPsec 节点间协商的安全服务策略的不同方面。这包括：

- ESP 加密算法
- 认证协议
- 密钥信息
- 密钥寿命
- SA 寿命
- AH 认证算法

当节点连接建立后，SA 就负责节点间的协商。在连接建立（及随后的重建立）过程中，每一个节点将它自己的安全参数索引（SPI）数年赋给它其它节点协商的 SA。SPI 是在节点之间交换的，并用于识别数据包。当一个节点接收到一个 IPsec 数据包，它会检查其中的 SPI，通过查找 SPI 数据库，找到相应的 SA，然后根据 SA 中的规则处理数据包。关于 ISAKMP 需要记住的一个关键点是它是独立于实现 IPsec 的密钥管理协议、密文和认证。

IKE 是 Oakley 密钥决定协议和 SKEME 密钥交换协议的混合体。IKE 协议负责管理 IPsec 节点的 ISAKMP 中的 IPsec 安全联系。IKE 协议可用于 ISAKMP，但它们并不是一样的。IKE 是建立 IPsec 节点间的 IPsec“连接”的机制。这要求对下面几种协商：

- 认证算法：IKE 使用 Diffie-Hellman 在非安全网络传输上建立共享的机密会话密钥。
- 机密性算法：IKE 节点将使用的安全性协议协商，它们是 AH、ESP 或 AH 和 ESP 组合。
- 哈希算法：IKE 使用哈希算法来验证报文数据。

- 认证密钥：IKE 支持使用预共享密钥、RSA 密钥（或者暂时）、数字认证和可扩展认证（Extended Authentication，Xauth)来进行认证管理。

IKE 可以在三种模式下操作：主模式、积极模式和快速模式。其中主模式和积极模式都实现相同的目标，即建立初始化阶段和第 1 阶段的 IKE SA。第一阶段的 SA 引导了 IKE 过程。一旦第一阶段的协商完成，快速模式就可以用于第二阶段的 IKE 操作，它允许全 SA 协商和在 SA 过期时刷新 SA 信息。

主模式、积极模式和快速模式之间的不同与安全级别和消息交换数量有关。主模式有六种信息交换模式（三种是在创建者中，三种是在响应者中）。主模式始于连接创建者发起一个保证 Diffie-Hellman 密钥交换安全的协商 SA。一旦协商的 SA 建立，用于快速模式认证、IKE 认证和 SA 加密的 Diffie-Hellman 密钥就会生成并交换，实现身份认证管理。

积极模式从连接创建者生成一个 Diffie-Hellman 密钥开始，目标是得到一个第一阶段的 SA 和节点的身份。然后响应者回复一个 SA 和身份认证数据，以及完成数据验证过程的创建者。整个的积极模式交换是在没有 SA 协商完成的，所有交换的数据都是不加密的。所以虽然与第 1 阶段有同样的交换信息，但一个更安全而另一个更快速。而且虽然快速模式与积极模式使用了相同数量的报文交换，但是快速模式更依赖在第 1 阶段协商中建立的身份认证和安全完整性。但至少现在我们应该很清楚 IPsec 的所有东西都是协商的。IPsec 节点间支持的安全服务是在两个节点建立通信时协商的。根据节点（如网关、主机）的不同类型，它会支持多个或一个 IKE 策略。当一个会话被初始化，连接的节点会发送它所有支持的 IKE 策略。远程节点会比较目标策略和它的最高优先级策略及按优先级从高到低顺序的后续策略，然后返回一个匹配的响应。只有两个节点都支持 IPsec 服务才能够在此运用。

例如，节点 A 能够支持数据加密和完整性验证服务，但节点 B 仅仅支持数据加密服务。因些节点 A 和 B 都需要有一个通用的 IKE 策略。这样，A 将需要有两个不同的 IKE 策略：一个支持数据和完整性服务的策略，还有一个仅支持数据服务的策略。为了使节点 A 和 B 能够通信，就只能使用数据加密服务。如果节点 A 仅仅有一个支持数据或完整性的 IKE 策略，那么 IKE 将会终止它们之间的协商，这样节点将不会建立一个 IPsec 连接。

（来源：TechTarget 中国　作者：Michael J. Martin　译者：曾少宁）

详解 SSL 协议

问：我经常听人们说 SSL 位于网络层和应用层之间，这是什么意思?

答：这是一个非常好的问题。我想回答这个问题的最好方法是从检查协议的目的开始。在计算机领域，协议是管理数据在两个端点之间传输的一套规则。这套规则包括连接、通讯和实时数据交换的句法、语义和同步执行。然而，大多数通讯和网络协议都不是单独发挥作用的，这些协议在一个称作协议堆栈的地方分层次地放在一起。协议堆栈是需要共同合作的协议的具体结合，在这个协议堆栈中的每一个协议都执行专门的任务。SSL(安全套接层)是一个基于标准的加密协议，提供加密和身份识别服务。SSL 广泛应用于在互联网上提供加密的通讯。SSL 最普通的应用是在网络浏览器中通过 HTTPS 实现的。然而，SSL 是一种透明的协议，对用户基本上是不可见的，它可应用于任何基于 TCP/IP 的应用程序。

正如你想象的那样，设法保证一个协议栈能够完成其预定的任务和保证不同的协议都在一起工作是非常复杂的。为了帮助工程师概念化协议栈，人们开发了各种模型，每一种模型都提供一个网络协议应该如何工作的简要说明。OSI(开放系统互连)模型可能是最被广泛应用的，它使用 7 层结构把各种协议以及服务组织在一起。早期的 TCP/IP 模型使用 4 层或者 5 层。这两种模型接近最上面的层在逻辑上更接近用户，接近最下面的层在逻辑上更接近数据的物理传输。

根据 OSI 模型，应用层(7 层)为应用程序处理提供普通的应用服务。网络层(3 层)解决把数据包从网络的一个地方传送到另一个地方的问题。SSL 是一种不同寻常的协议，它不只在一个层上工作。SSL 既不是网络层协议也不是应用层协议，它是位于这两层之间的一种协议。

由于 SSL 所处的位置，SSL 能够向客户机提供有选择地保护单一应用程序的能力，而不是对整个一组应用程序进行加密。这个过程能够在不用担心 3 层(网络层)的情况下完成。由于这些原因，当使用 SSL 对网络通讯进行加密的时候，实际上只加密了应用层数据。这与 IPsec 协议不同。IPsec 协议在网络层工作，加密在 IP 层中的所有通讯数据。

(来源：TechTarget 中国　作者：Michael Cobb 译者：王菲)

黑帽大会 2010：SSL 协议使用的最新测试细节

安全研究人员 Ivan Ristic 一直在默默地调查数百万个注册域名，以查明和测试 SSL 协议的实施情况。

Ristic 是 Qualys 公司的工程部总监，负责 SSL 实验室的管理工作。该实验室从事非商业性的研究工作，于去年被 Qualys 公司收购。该实验室的网站利用 SSL 测试工具检查配置问题和协议错误，而这些缺陷有可能被中间人攻击（man-in-the-middle attacks）所利用，诱骗人们交出自己的敏感数据。

“我们正试着在互联网上找到尽可能多的 SSL 服务器，并对它们一一进行评估”，Ristic 说，“我们的目标是弄明白 SSL 真正的使用现状。我们需要知道我们是否安全，如果存在安全隐患，我们还需要知道这些安全问题究竟是什么，可以采取哪些措施来解决这些问题。”

Ristic 计划于本月下旬在 Las Vegas 举办的 Black Hat 大会上详细陈诉 SSL 目前的研究细节。Ristic 表示，在审查的约 1.2 亿个域名中，约 72 万个域名使用了 SSL 认证。在本次采访中，Ristic 解释了研究人员需要对 SSL 进行重点关注的原因，尽管大家都认为它是一个近乎完美的安全协议。

请介绍下 SSL 实验室，它是怎样成为 Qualys 公司一部分的呢？

Ristic：SSL 实验室是一个重点关注 SSL 和 TLS 协议的研究组织，是我在大约一年前创立的，这来源于我对 SSL 的狂热爱好。当我发现 SSL 是一个如此优秀的协议时，我对 SSL 便着了迷。SSL 是最成功的协议之一，是网络安全的支柱，可我们在研究它的使用情况、帮助各地的用户配置 SSL 以及正确使用 SSL 等方面花的时间却很少。在我所创建的这个网站上，有大量可以帮助人们了解如何正确配置 SSL 的信息和工具。Qualys 公司非常看重我所做的研究。在收购之前，我可以说是一边负责 SSL 实验室，一边做其他的事情。而 Qualys 公司给我提供了一次机会，使我能够把精力全部放在这项研究上。

为什么缺乏对 SSL 的研究？

Ristic：我认为其中的原因是，SSL 在最初阶段就取得了许多令人激动的研究成果。该协议自创立以来大约有 15 年了。多年来，不知道什么原因，我们没有重视 SSL，反而研究应用安全领域、甚至是网络安全领域等其他方面的问题去了。我觉得，人们过去对 SSL 有一种约定俗成的看法，那就是没有必要对它进行认真思考，这不得不说是一件令人惋惜的事。不过，几年下来，这种情况有了比较明显的改观。关于 SSL，我们已经有了不少新的认识。有几个相互独立工作的研究人员发现，SSL 不仅在实施过程中存在许多问题，其协议本身也存在一些需要

解决的细小问题。现在，对 SSL 的研究又成了一个热点，越来越多的人开始不再沉默，并致力于解决 SSL 的问题。

许多问题都是源于配置吗？

Ristic：是的。实施工作中存在问题，协议本身也存在一些小问题。如果您是一名普通用户，您不必对协议问题关注太多。这是 SSL 厂商需要关心的问题，是研究人员改善该协议需要解决的问题。作为一个普通用户，您只需要保持系统更新。如果您保持了系统更新，那么您就可以应对这些问题了。至于配置问题，您可以马上对其进行了解，并且在半小时或几个小时内就能把问题解决。然而，大多数人根本没有意识到他们的配置存在问题。

配置是在线 SSL 评估工具的基础吗？所有这些问题都需要检查一遍吗？

Ristic：是的。在线工具分为两部分：第一部分是系统方法，它详细阐明了如何进行 SSL 评估；第二部分是工具。如果您键入了网站域名，该工具将转到该域名，并找到后台所有的 SSL 服务器，甚至是相同域名下的多个 SSL 服务器，然后对服务器逐一进行评估。评估的结果很容易看懂，因为我们对每种网站的配置方式都进行了总结，分成 A 至 F 级并用 0 到 100 进行编号。所以，即使您没有深入研究 SSL，也会很容易明白。而如果您是技术用户，我们还会提供更多的技术信息。这两类用户对此都感到满意。

您曾经谈到过 SSL renegotiation 漏洞。这是一个什么问题？

Ristic：renegotiation 漏洞是去年年底发现的。基本上，这个问题来自 SSL 的某一特殊层面，这是 SSL 的优势也是劣势。SSL 被设计成协议无关的，所以它位于一个独立的网络层。这使得它适用于任何底层的网络协议。您可以使用 SSL 保护 HTTP、电子邮件的 SMTP 协议、IMAP、LDAP 以及其他协议。而要想让这些不同的协议都与 SSL 一起使用，几乎不需要做什么工作。但是，我们现在明白了，当您部署一个没有直接连接到 SSL 的协议时，威胁就会被引入。最终，我们将被迫面对不匹配问题，以及其他一些您可能无法处理的 SSL 问题，因为您对其完全不了解。其本质是允许攻击者打开两个连接，并诱使用户把这两个连接当成一个来使用，由此攻击者便可以向有漏洞的网站引入任意内容。这一点修复起来相对快速，但实际上现在的问题是，所有的 SSL 服务器都需要打补丁。在对正在打补丁的无数服务器进行调查时，该漏洞便是我要检查的重点之一，以便了解当系统问题出现时管理员的反应速度有多快。

您的测试是用 2000 个数据包对单个服务器进行快速测试吗？

Ristic：这项评估的设计在构想之初是打算包含大量评估测试的，但在没有一个完整的 SSL 连接的情况下，只能在一个非常低的层次进行实施，不过这也使得速度非常快。一次测试需要进行约 200 次连接，数据交换的速度为 250 kbts。我们不希望所测试的服务器超负荷运

转，不过测试本身并不会损害任何服务器。我们正试着尽可能多的找到互联网上的 SSL 服务器，并对它们一一进行评估。我们的目标是弄明白 SSL 真正的使用现状。我们需要知道我们是否安全，如果存在安全隐患，我们还需要知道这些安全问题究竟是什么，可以采取哪些措施来解决这些问题。”

网站有数百万个，但只有相对较少的网站使用了 SSL 认证。这是否属实？

Ristic：是的。总共约有 2 亿个注册域名，在测试中我调查了其中的 60%（1.2 亿个）。弄明白所调查的 1.2 亿个网站都支持哪些服务，只是我工作的开始。我还想了解每个域名所代表的网站是否都真实存在，所运行的网站又是否安全。最终我们发现，约 1/4 的域名都无法访问。在所测试的域名中，约 77%有一个服务器。大约 2200 万个域名有 Web 服务器，约 3%的域名可能具备有效的 SSL 认证。我还发现，大约有 72 万个网站是安全的，在测试的第二阶段我还将对其进行更深入的评估。目前，我们在 SSL 领域面临着一个重大的问题：如果托管一百万个网站，您可以把它们全放在一个 IP 地址上；可对于每一个安全的网站来讲，您只能拥有一个独立的专用 IP 地址。这一问题很难解决，阻碍了 SSL 在世界范围内的普及。

您已经参与了开源 Web 应用程序防火墙 ModSecurity 项目。对于 Trustwave 收购 Breach Security 以及 ModSecurity 项目，您作何反应？

Ristic：收购可能会影响 ModSecurity 项目，但我们并不知道他们以后对 ModSecurity 有什么打算。说实话，ModSecurity 项目在 Breach Security 下并没有良好发展，所以我认为收购可能会让情况变得更好一些。虽然他们在维护 ModSecurity 项目上做得的确很好，但在过去几年里我们没有看到 ModSecurity 取得过什么进展。我仍然会以贡献者的身份参与此项目，如果有时间还将继续作出自己的努力。

为什么 ModSecurity 项目的进展不大？

Ristic：我认为是因为 ModSecurity 与 Breach Security 的利益不一致造成的，因为 Breach Security 除了 ModSecurity 还有他们自己的产品。不幸的是，在 ModSecurity 被收购后，Breach Security 并没有将其融入到自己的产品线中。如果当初他们做到了这一点，ModSecurity 会因此而受益，因为 ModSecurity 也会成为 Breach Security 利益的一部分。

（来源：TechTarget 中国　作者：Robert Westervelt　译者：sean）