TCP/IP网络中隐蔽信道的实现和特点

张敏　张彤

西北核技术研究所　陕西　710024

摘要：网络隐蔽信道是网络攻击中的一种重要手段，是对网络安全的一大威胁。本文介绍了TCP/IP网络中隐蔽信道的多种实现方法，对每一种隐蔽信道，分析了带宽等性能特点，以及信道的改进方法和限制措施。

关键词：网络安全；隐蔽信道；TCP/IP协议

0 引言

隐蔽信道是一种能抵抗审查的通信方式，有效的隐蔽信道不会被对手觉察到信道的存在。和加密不同，隐蔽信道是要对通信本身加以隐藏，让对手根本不知道该信道的存在，而加密是保护通信的内容，对手在截获信息的情况下也不能得知通信的内容。

隐蔽信道的存在是对计算机系统安全的很大威胁，木马注入到被入侵计算机，很多时候需要和攻击者进行通信，以便将获取的信息传给攻击者，或者接受控制指令。受到网络访问控制系统(NACS)监视的情况下，新开一条连接建立信道是不现实的。这时隐蔽信道是一个理想的选择，在正常网络通信中实现隐蔽信道，这一点正好满足木马程序的必备要素之一：隐蔽性。隐蔽信道也为当前的分布式拒绝服务攻击(DDoS)提供了机会和方法。没有隐蔽信道，攻击者无法控制用于发起攻击的分布代理，如果消除了与分布代理的通信能力，那么就消除了分布式攻击的威胁。

计算机网络中的隐蔽信道主要是利用协议中的冗余条件，在现有网络通信中“寄生”隐蔽通信。本文主要分析TCP/IP网络中的隐蔽信道。

1 网络分层模型

计算机网络是分层的，常见的是五层协议的体系结构：应用层、传输层、网络层、数据链路层、物理层。最上层的应用层将各种应用程序与网络服务联系起来。传输层则负责端到端可靠和透明的数据传输，这一层主要有UDP和TCP两个协议。网络层为数据包在网络中的传输提供寻址和路由功能。再下面的数据链路层和物理层负责与实际网络设备交互。

在协议栈的各层中都可能实现隐蔽通信。网络层以下的协议数据(比如MAC地址)在跨网段时会发生变化，所以在这些层中的隐蔽信道只能局限在同一网段中，本文不作介绍。后面章节中将详细分析目前广泛使用的TCP/IP协议中能够构造出的各种隐蔽信道。并且将分析信道的性能特点：

带宽：能传输的隐蔽信息的bit数；

可检测性：信道被检测到的难易程度；

使用环境：在什么样的条件下可用；

阻止难度：如何阻止隐蔽信道的使用。

2 网络层隐蔽通信

网际协议IP是TCP/IP体系中两个最主要的协议之一，也是最重要的因特网标准协议[RFC791]之一。在IP协议中隐蔽信道的构造方法主要是对数据包头字段的操作。

一个IP数据包由包头和数据两部分组成。包头的前一部分是固定长度，共20字节，是所有IP数据包必须具有的(图1所示)。各字段中适合于构造隐蔽信道的有服务类型、标识、标志、片偏移、源地址、目的地址等。

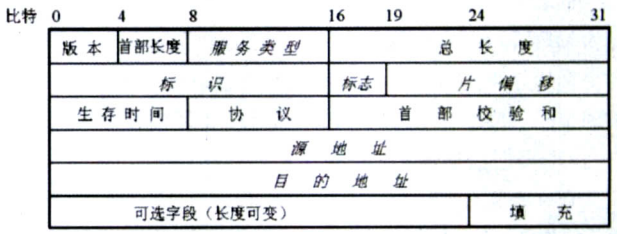

图1　IPv4数据包头

2.1 服务类型

服务类型字段占8bit，用来获得更好的服务。服务类型的8个比特均可用于传输隐蔽信息，但对8个比特都进行编码会更大地改变实际通信流量，降低隐蔽性，更容易被检测到，特别是未用的第7比特也用于编码时，因为此bit一般应设为0。为此，可以仅使用其中的某些位，如D位，但这样会减小信道带宽。

该信道带宽最大可到每IP包8bit，某些bit不编码时会减小信道带宽。在相当长一段时间内并没有使用服务类型字段，直到最近需要将实时多媒体信息在因特网上传送时，该字段才重新引起重视。因此，使用该字段进行隐蔽通信容易引起怀疑。消除该信道只需要重置服务类型字段为0。

2.2 标识

标识占16bit，为了保证被切分的数据包接收方能正确地重组，由数据包发送者设定的鉴别值。当数据包由于长度超过网络的最大传输单元(MTU)而必须分片时，这个标识字段

作者简介：张敏(1981-)，男，硕士研究生，研究方向：通信保密与网络安全。
张彤(1967-)，男，研究员，博士，研究方向：网络安全和信息隐藏技术。

的值就被复制到所有分片的标识字段中。相同的标识字段值使得各分片最后能正确地重组成为原来的数据包。因此当数据包处于网络中时，该字段的值必须是惟一的。RFC791中没有具体的实现，操作系统按自己的方法生成该字段值。

文献对在该字段嵌入隐蔽信息有详细描述，作者对该字段的高8位编码，将要嵌入的值(ASCII码)加密处理得到8bit值作为标识的高8位，低8位则随机产生，以使整个字段更具随机性。这种编码方式可以产生每数据包8bit的带宽。但由于各操作系统产生的标识值都有相应的特征，这使得检测该信道成为可能：

(1) 标识值全局递增，某些操作系统特别是比较早期的(如linux<2.4)使用一个全局计数器来产生这一字段值。

(2)标识值主机递增，一些操作系统(如linux>=2.4)为每一主机使用一个计数器以生成标识值。

(3) 标识值的 MSB 绑定，OpenBSD 在每一个周期(3 分种或每30000个标识)都固定标识的MSB(most significant bit)。

(4)每一周期中，OpenBSD 操作系统中标识都不会重复。

熟悉操作系统中标识的实现方法的情况下，这些异常都可能会导致隐蔽通信失败。

2.3 标志

标志位(flag)占3bit，目前只有前两个bit有意义，最低位记为MF(More Fragment)，MF=1即表示该数据包分片后面还有更多的分片，MF=0表示这已是数据包最后一个分片。标志位的中间一位是DF(Don 't Fragment)，即不能分片标志，只有当 DF=0 时数据包才允许分片。

在标志位的所有3bit中都嵌入隐蔽信息是不明智的，因为第一位为保留位，默认必须为0。后面两位的利用情形可以是：

发送者了解网络 MTU 的情形下，数据包长度小于 MTU，这种环境下无论如何数据包是不会被分片的，那么DF可以设置为0也可以设置为1，这个bit的使用并不会影响包分片的重组。发送者不了解网络 MTU 时，只能将 DF 置 1，即不允许分片，而对 MF 进行编码。

两种情况下带宽都是较小的，只有1bit每数据包。而一般情况下，DF 位为 1 时，MF 必须为 0，所以较可靠的使用方法还是在发送者了解 MTU 的情况下只编码 DF 位，MF 置 0。

2.4 片偏移

数据包分片后，某分片的片偏移字段指出该片在原数据包中的相对位置，保存的是该分片中数据部分首字节在原数据中的相对位置，片偏移以8个字节为偏移单位。接收方利用该字段和 MF、DF 标志可以将具有相同标识值的分片正确地重组出原数据包。

片偏移字段共8bit，能够利用的情况下可以编码8位，但更合理的作法是将其值限定在数据长度的范围以内，这却限定了隐蔽信道的带宽。DF 为 1 时不允许分片，片偏移的值不影响分片；DF 为 0 且发送者了解 MTU 以保证分片不会发生的情况下，也可对片偏移字段编码。

发送者了解网络 MTU 的情况下，也可以对数据包分片进行正常的切分，在切分时控制分片长度，以便在片偏移字段中嵌入隐蔽信息。这种方法更具合理性，不会产生很大的异常现象。

2.5 源地址

IP数据包头中的源地址字段32bit，保存的是源IP地址(RFC791)。利用此字段构造隐蔽信道，即伪造源 IP 地址，由于路由器一般情况下只允许源IP地址属于本地网段内的数据包出去，这种方法的编码空间必须限制在本地网段内。比如10.1.1.0/24网段内，可以编码的只能是最后8bit，除去全0、全1，可以伪造的IP地址范围是10.1.1.1~10.1.1.254，如果需要隐蔽传输ASCII字符，只需将字符对应的ASCII值放在IP地址的末8位以生成伪造的IP源地址，并将数据包的目的地址设为某一特定值(或使用其他同步方法)，以指示接收端(可能是中间路由器)接收隐蔽信息。

这一方法的带宽依赖于可编码地址空间的大小，源IP地址中用于网段内主机的bit数越多，就能够嵌入更多的信息。但是网络出口存在的IP-MAC绑定校验可能导致该方法失效。

2.6 目的地址

IP数据包头中的目的地址字段32bit，保存的是目的IP地址(RFC791)。也可以采用3.5类似的方法，以实现网段内某主机和某路由器(如出口路由器)的隐蔽通信，该主机需要传输隐蔽信息时(采用某种事先约定好的同步方法)，选择M个目的地址中的一个(设为第K个)作为目的地址构造数据包，路由器收到该主机的数据包时读取到目的地址，其值为 M 个目的地址中的第 K 个，那么这个 K 即为隐蔽信息。

信道带宽可达到log2M bit。消除此隐蔽通信，涉及到网络流量分析攻击。

3 传输层隐蔽通信

传输层为应用进程之间提供端到端的逻辑通信(而网络层是为主机之间提供逻辑通信)。TCP 报头格式如图 2 所示[RFC793]，其中适合用于隐蔽信道的字段有：序列号、确认号、选项、窗口字段。

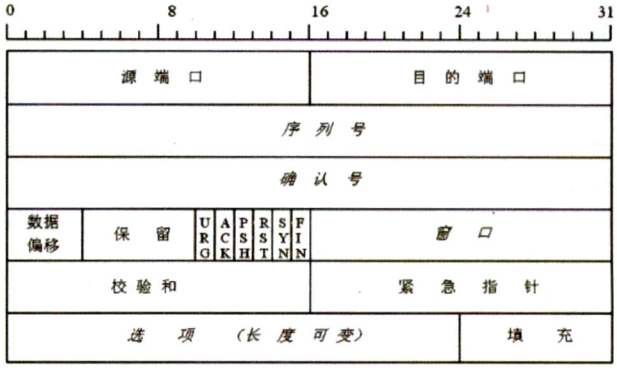

图 2 TCP 数据包头

3.1 序列号

TCP是面向数据流的，TCP传送的报文可看成为连续的数据流，TCP把在一个TCP连接中传送的数据流中的每一个字节都编上一个序列号。整个数据的起始序列号在连接建立时设置。使用序列号传递隐蔽信息一般是编码初始序列号(ISN)，即SYN包中的序列号。

A要发送信息x到B时，将x进行编码生成32bit的数值Sx，将这个数值作为TCP包的序列号，并将SYN位置1，表示这个TCP包为连接请求包。然后发出包，B收到A发来的连接请求包，读取到Sx，再解码出x。

这种隐蔽信道实现起来简单，带宽最大可达每个SYN包32bits。但是每种操作系统产生的ISN都有其非常明显、非常严格的结构特征，比如：

(1)最高字节(most significant byte)的周期性，Linux2.2和早期2.4中，ISN的most significant byte被设为当前时间(秒)除以300，linux2.6和后期2.4中，ISN的most significant byte被设为系统启动后运行时间(秒)除以300，因此ISN的MSB以5分钟为周期递增。

(2) OpenBSD产生的所有ISN的第15bit均设为0。在每一个周期(3分种或每30000个标识后)都固定ISN的MSB(most significant bit)。每一个周期间隔中，OpenBSD产生的ISN的第16至30位都不重复。

若非熟悉操作系统的ISN实现，一般经过修改的ISN很容易被监视者发现，这当然需要监视者本身熟知操作系统实现ISN的产生方法。根据操作系统实现方法，采取极为细致的编码，能够提高信道的生存能力，同时却会降低带宽。

3.2 确认号

确认号是接收端期望收到对方下一个文段的数据的第一个字节的序号，也就是期望收到的下一个数据报头部的序列号字段的值。

基于确认号的隐蔽信道利用了TCP协议的三次握手：

(1) X需要和Y建立TCP连接时，X发起SYN包(SYN置1)到Y，其中的序列号为x：

X--------SYN,SEQ=x-------->Y

(2) Y收到后确认此包，回应SYN/ACK包(SYN,ACK均置1)，其中的序列号y，确认号x+1：

X<-------- SYN/ACK, SEQ=y, ACK=x+1--------Y

(3) X收到确认包后再确认此包，回应ACK包(ACK置1)，其中序列号x+1，确认号y+1：

X-------- ACK, SEQ=x+1, ACK=y+1-------->Y

经过三次握手后连接建立。

X欲与Y发送隐蔽信息x时，X构造SYN包，SEQ设为x，并将包的源IP地址设为Y，目标IP地址设为Z，其中Z为“跳板”：

Y--------SYN,SEQ=x-------->Z

Z收到此SYN包，发现是Y发来的连接请求(因为源地址是Y)，根据三次握手协定，向Y回应SYN/ACK包，并将确认号设置为x+1

Y<-------- SYN/ACK, SEQ=z, ACK=x+1--------Z

Y发现自己并没有发起SYN包却收到Z发来的SYN/ACK包，即知道是X送来了隐蔽信息，读取ACK值，减去1得到x。

ACK域为32位，所以这种隐蔽信道带宽为每个SYN包32bits。这种方法只是X到Y的单向通信，但它可以将信息的发送者(X)隐藏起来，因为在Y端看来，数据包是从“跳板”Z回应过来的。

3.3 选项

timestamps是TCP选项的一种，用于提供更好的服务。时间戳选项在RFC1323有详细的描述：

表1 timestamps选项格式

字节	1	1	4	4
含义	Kind=8	10	TS Value (TSval)	TS Echo Reply (TSecr)

第1字节是TCP选项类型，值为8表示该选项为时间戳。第2个字节为选项长度，等于10，通常我们将发送的时间戳和应答的时间戳集合在一起，形成了以上的形式，总长度是10bytes。选项中有两个4bytes的时间部值，TSval是TCP发送此选项时的时钟值。TSecr仅出现在ACK被置位的情况下，它显示一个由远端TCP发送的时间域TSval的值。

用于编码传递隐蔽信息的一般是TSval。timestamps必须是单调递增的，因为后面的时间总是大于前面的时间，所以对timestamps的修改必须细致，不能产生非法时间。timestamps还必须反映实际时间，其值必须大致等于当前系统时间，要不然会产生异常，导致隐蔽通信可能被监视者发现，所以一般只能对timestamps的最低位进行编码。同时，修改后的timestamps还必须与未经修改的timestamps特征上一致，比如慢速网络流量中，正常timestamps的最低位是随机的。

4 总结

网络中存在各种各样的隐蔽信道，正确地理解它们的工作原理及性能特点，对于隐蔽信道构造者来讲，可以改进隐蔽信道的实现方法，更加合理的控制信道带宽，以便逃过网络访问控制系统的监控；对于安全监管人员来讲，可以改进网络访问控制系统，以便能准确地检测到隐蔽信道，并采取有效的措施消除或审计检测出的隐蔽信道。

参考文献

[1]http://www.cyberguard.com/download/white_paper/en_cg_covert_channels.pdf.

[2]谢希仁编.计算机网络(第4版).电子工业出版社.2003.1.

[3]C.H.Rowland.Covert channels in the TCP/IP protocol suite. Peer Reviewed Journal on the Internet.1997.