网络安全——

网络攻击—拒绝服务攻击

北京邮电大学

郑康锋

zkfbupt@163.com

什么是DoS?

Denial **o**f **S**ervice, 拒绝服务

拒绝服务攻击——

大家熟悉的DoS？

典型的拒绝服务攻击

- Ping of Death
- Teardrop
- Land
- Syn Flood
- Smurf
- HTTP Flood
- CC

典型DoS—Ping of Death

- ping is a computer network administration software utility used to test the reachability of a host on an Internet Protocol (IP) network.

- It measures the round-trip time for messages sent from the originating host to a destination computer that are echoed back to the source.

- **Ping operates by sending Internet Control Message Protocol (ICMP/ICMP6) Echo Request packets to the target host and waiting for an ICMP Echo Reply.** The program reports errors, packet loss, and a statistical summary of the results, typically including the minimum, maximum, the mean round-trip times, and standard deviation of the mean.

典型DoS—Ping of Death

The ***echo request*** ("ping") is an ICMP/ICMP6 message.

00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
Type = 8(IPv4, ICMP) 128(IPv6,ICMP6)								Code = 0								Header Checksum															
Identifier																Sequence Number															
Payload																															

The *echo reply* is an ICMP message generated in response to an echo request.

00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
Type = 0(IPv4,ICMP) 129(IPv6,ICMP6)								Code = 0								Header Checksum															
Identifier																Sequence Number															
Payload																															

典型DoS—Ping of Death

- A ping of death is a type of attack on a computer system that involves sending a malformed or otherwise malicious ping to a computer.。
- 早期的操作系统处理ICMP分组时，只开辟64KB缓冲区用来存放收到的数据包。
- 攻击者故意在ICMP Echo数据包(Ping包)之后附加非常多的冗余信息，使数据包的尺寸超过65535个字节的上限。
- 接收方对这种数据包进行处理时就会出现内存分配错误，导致TCP/IP堆栈溢出，从而引起系统崩溃，挂起或重启。

```
C:\Users\bupt>ping -l 66000 192.168.0.201
选项 -l 的值有错误，有效范围从 0 到 65500。
```

典型DoS——Teardrop

- A teardrop attack involves sending mangled IP fragments with overlapping, oversized payloads to the target machine.
- This can crash various operating systems because of a bug in their TCP/IP fragmentation re-assembly code.[54] Windows 3.1x, Windows 95 and Windows NT operating systems, as well as versions of Linux prior to versions 2.0.32 and 2.1.63 are vulnerable to this attack.
- One of the fields in an IP header is the “fragment offset” field, indicating the starting position, or offset, of the data contained in a fragmented packet relative to the data in the original packet. If the sum of the offset and size of one fragmented packet differs from that of the next fragmented packet, the packets overlap.

From Wikipedia

典型DoS——Teardrop

Fragmentation and reassembly

Fragment	Total bytes	Header bytes	Data bytes	"More fragments" flag	Fragment offset (8-byte blocks)
1	2500	20	2480	1	0
2	2040	20	2020	0	310

an MTU of 1,500 bytes， For example, consider a Transport layer segment with size of 4,500 bytes.

Fragment	Total bytes	Header bytes	Data bytes	"More fragments" flag	Fragment offset (8-byte blocks)
1	1500	20	1480	1	0
2	1020	20	1000	1	185
3	1500	20	1480	1	310
4	560	20	540	0	495

We can use the last offset and last data size to calculate the total data size: 495*8 + 540 = 3960 + 540 = 4500.

https://en.wikipedia.org/wiki/IPv4#Fragmentation_and_reassembly

典型DoS——Teardrop

- MTU（maximum transfer unit，最大传送单位）限制传输数据的包大小，大数据包需要分段。Teardrop攻击就是利用这种分割重组间的漏洞而产生的攻击方式。

- Teardrop指的是向目标机器发送损坏的IP包，诸如重叠的包或过大的包载荷，该攻击通过TCP/IP协议栈中分片重组代码中的bug来瘫痪各种不同的操作系统。

- Teardrop攻击使接收数据方重组数据包时，出现数据包长度超大（如负值），导致溢出。(假设数据包中第二片IP包的偏移量小于第一片结束的位移，而且算上第二片IP包的Data，也未超过第一片的尾部，这就是重叠现象。)

典型DoS——Teardrop

第一片：Fragment offset=0；数据包长度（length）=ip.total-length-ip.headerlength=36。

IP 首部(20 字节)	UDP 首部(8 字节)	数据 1(28 字节)

第二片：Fragmentoffset=32；数据包长度（length）=ip.total-length-ip.headerlength=3。

IP 首部(20 字节)	数据 2

若两片重组，则第二片必然嵌于第一片内，

1	8	9	32	35	36
IP 首部(20 字节)	UDP 首部(8 字节)	数据 1	数据 2		

这时第一片的 end=36，offset=0；第二片的 end=32+3=35，而正常情况下第二片的段偏移 offset=36，这样造成 fp->len=end-offset=35-36=-1。此时调用 memcpy ((ptr + fp->offset), fp->ptr, fp->len)时，由于 fp->len 为负数，会引起堆栈溢出。

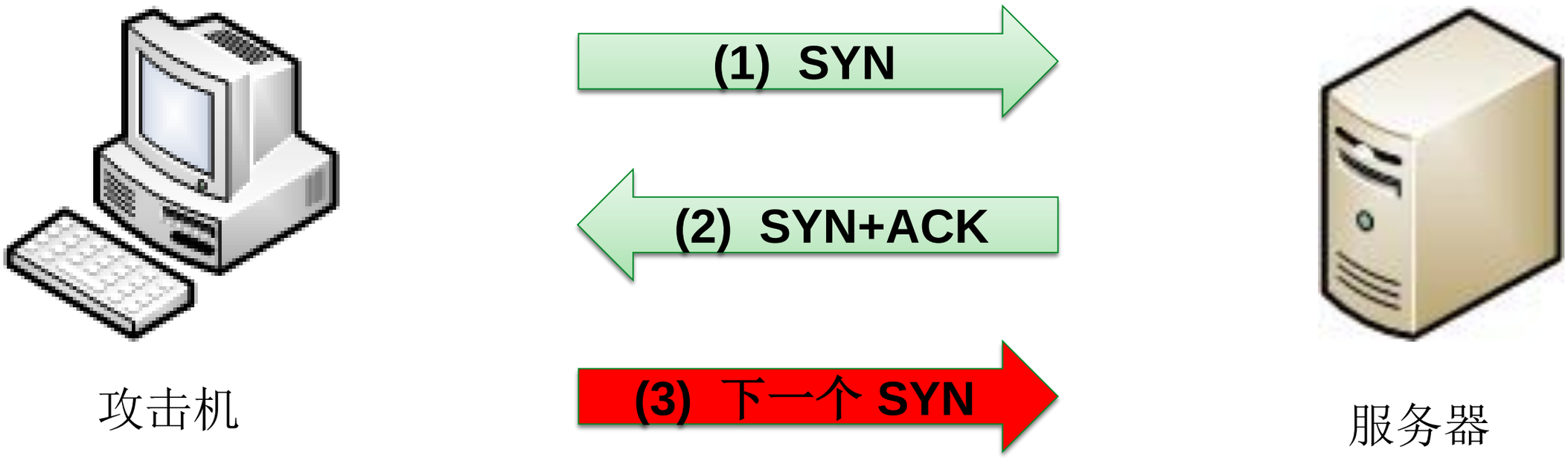
来自：DoS攻击原理与对策的进一步研究 陈波 于泠

典型DoS——Land攻击

- Land攻击：利用特殊的TCP封包传送至目标主机，使其因无法判别而当机或被迫重新启动。
- 攻击原理是：用一个特别打造的SYN包，它的源地址和目标地址都被设置成某一个服务器地址。此举将导致接受服务器向它自己的地址发送SYN一ACK消息，结果这个地址又发回ACK消息并创建一个空连接。被攻击的服务器每接收一个这样的连接都将保留，直到超时。

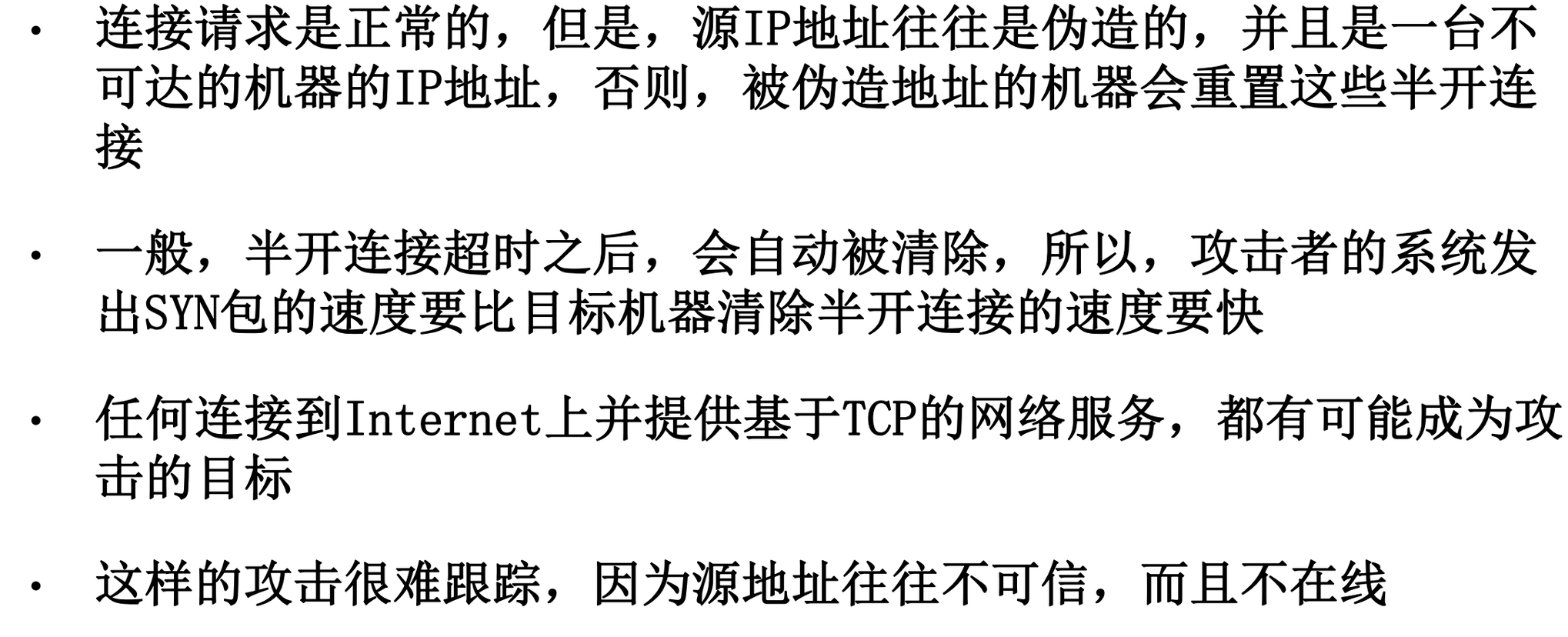
典型DoS—SYN洪水

- SYN洪水
  - 原理：
    - 每个机器都需要为半开连接分配一定的资源
    - 这种半开连接的数量是有限制
    - 共计方利用TCP连接三次握手过程，打开大量的半开TCP连接
    - 目标机器不能进一步接受TCP连接。机器就不再接受进来的连接请求。



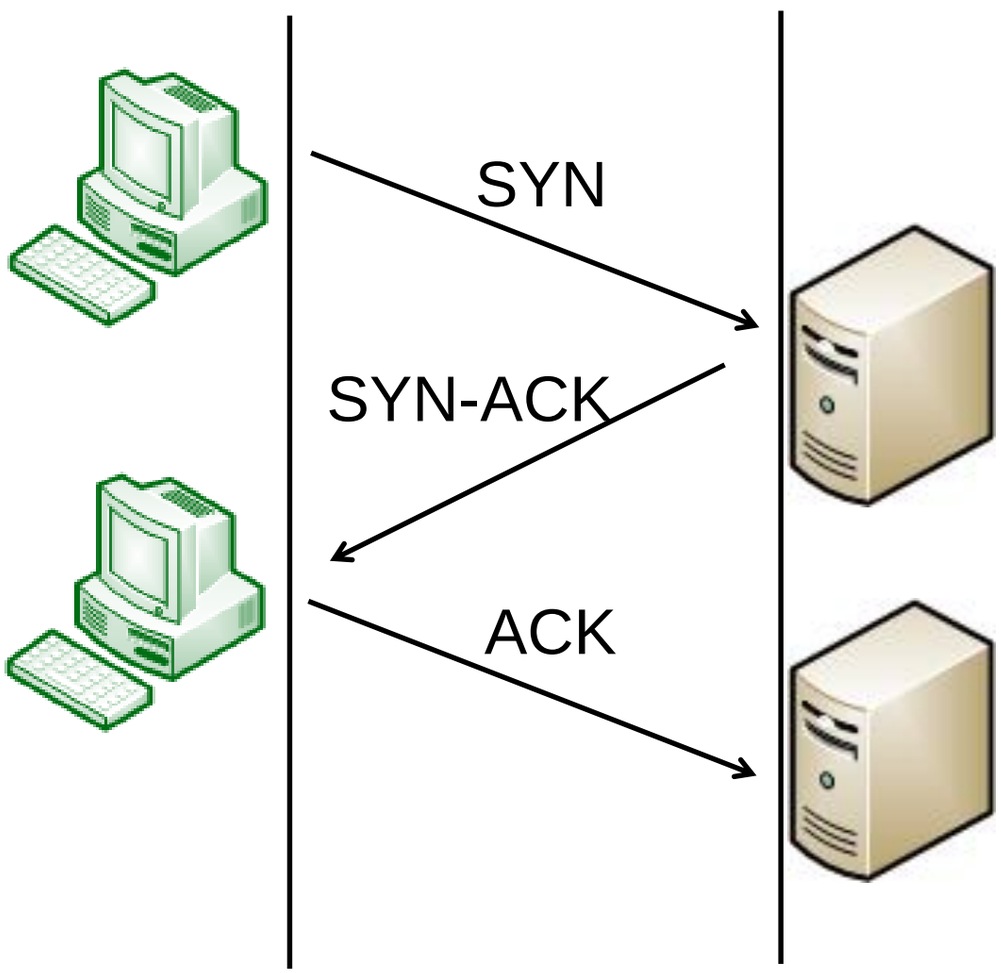
典型DoS—SYN洪水

- SYN洪水
  - 受影响的系统：大多数操作系统
  - 攻击细节

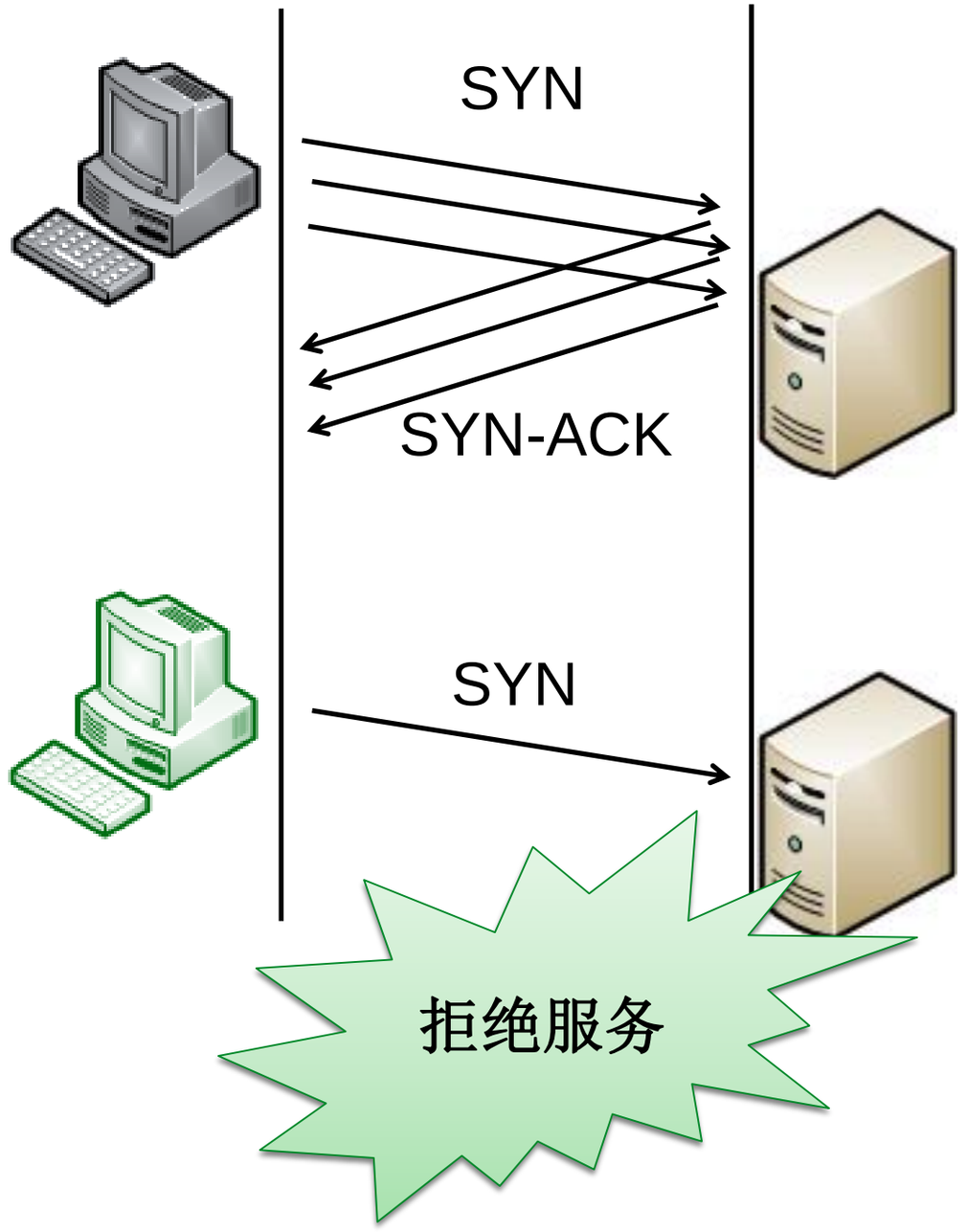


典型DoS—SYN洪水

正常的SYN连接：



SYN/ACK Flood Attack：

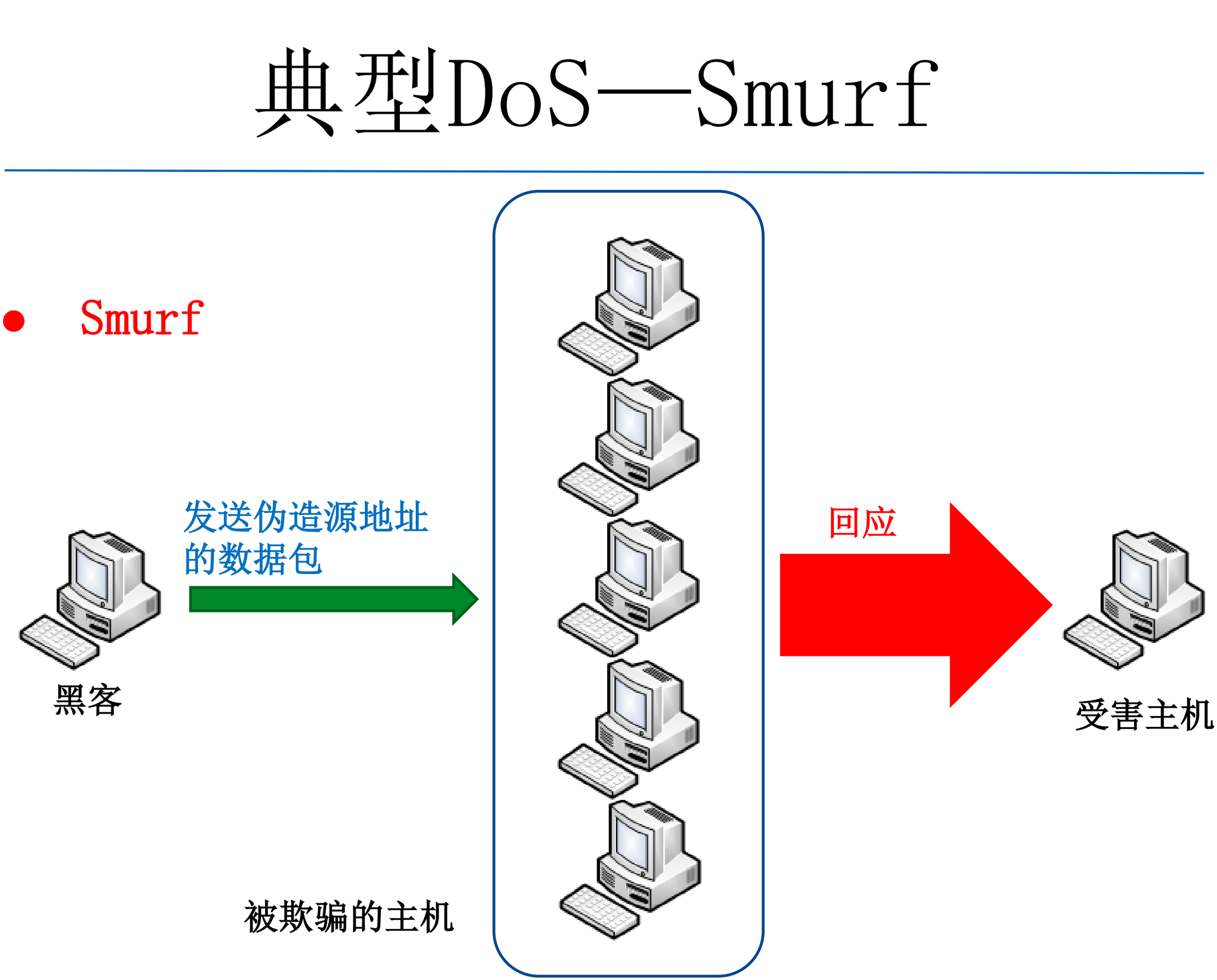


典型DoS—Smurf

- Smurf
  - 原理：
    - 攻击者向一个广播地址发送ICMP Echo请求，并且用受害者的IP地址作为源地址
    - 广播地址网络上的每台机器响应这些Echo请求，同时向受害者主机发送ICMP Echo-Reply应答
    - 受害者主机会被这些大量的应答包淹没
  - 受影响的系统：大多数操作系统和路由器

典型DoS—Smurf

- Smurf



典型DoS—HTTP洪水

- HTTP洪水
- 这类攻击会占用大量的HTTP进程，从而耗费大量的系统资源。最终，会导致系统因不堪重负而崩溃掉。
- 以最典型的HTTP GET FLOOD攻击为例。
  - HTTP GET FLOOD是针对应用服务器上的某个文件，对其进行快速的反复的重复读取操作，从而造成服务器的资源减少直至崩溃。
  - HTTP GET FLOOD针对的不仅仅是WEB服务器，还有数据库服务器。大量的HTTP请求产生了大量的数据库查询，可以在几秒之内使数据库停止响应，系统负载升高，最终导致服务器宕机。

传统DoS的简单变化

CC攻击：ChallengeCollapsar

- 原理：CC攻击的原理就是攻击者控制某些主机不停地发大量数据包给对方服务器造成服务器资源耗尽，一直到宕机崩溃。

- 特点
  - 真实度高
  - 检测困难
  - 防御困难

小讨论：如何检测CC攻击？

传统DoS的简单变化

慢速攻击

- HTTP Post慢速DoS攻击：2012年的OWASP大会上，由Wong Onn Chee 和 Tom Brennan共同演示。
- 基本原理：对HTTP服务器，先建立了一个连接，指定一个比较大的content-length，然后以非常低的速度发包，比如1-10s发一个字节，然后维持住这个连接不断开。
- 种类
  - Slow headers：Web应用在处理HTTP请求之前都要先接收完所有的HTTP头部，因为HTTP头部中包含了一些Web应用可能用到的重要的信息。攻击者利用这点，发起一个HTTP请求，一直不停的发送HTTP头部，消耗服务器的连接和内存资源。
  - Slow body：攻击者发送一个HTTP POST请求，该请求的Content-Length头部值很大，使得Web服务器或代理认为客户端要发送很大的数据。
  - Slow read：客户端与服务器建立连接并发送了一个HTTP请求，客户端发送完整的请求给服务器端，然后一直保持这个连接，以很低的速度读取Response，比如很长一段时间客户端不读取任何数据，通过发送Zero Window到服务器，让服务器误以为客户端很忙，直到连接快超时前才读取一个字节，以消耗服务器的连接和内存资源。

拒绝服务攻击——

DoS检测

DoS检测

- 按检测模式分类的检测方法
  - 基于误用的DDoS检测
  - 基于异常的DDoS检测
  - 混合模式DDoS检测
- 按算法部署位置分类的检测方法
  - 源端检测
  - 中间网络检测
  - 目的端检测
  - SDN下的DoS检测

基于误用的DDoS检测

- 基于误用的DDoS攻击检测是指事先收集已有DDoS攻击的各种特征，然后将当前网络中数据包的特征和各种攻击特征相互比较，如果特征匹配则发现DDoS攻击。
- 基于误用方法依赖于**攻击特征的选取**，一般用于检测利用漏洞型的DDoS攻击。
- 基于误用的DDoS检测主要是利用了**特征匹配**、**模型推理**、**状态转换**和**专家系统**的方法。
  - 特征匹配主要是利用各种DDoS的特征进行检测。
  - 模型推理也是利用DDoS攻击的特征进行检测。
  - 状态转换将DDoS攻击看成被系统监测的一系列系统状态转换和相对应的条件，攻击事件与系统状态不要求一一对应。
  - 专家系统将专家关于DDoS攻击检测的知识转换成特征库中的特征与规则。检测系统一旦认为网络中出现了与专家系统中攻击发生的条件相匹配的现象，就判定发生了攻击。

基于异常的DDoS检测

- 基于异常的DDoS攻击检测是指通过监视系统审计记录上系统使用的异常情况，可以检测出违反安全的事件。目前，大多数的DDoS攻击检测都属于异常检测。

- 基于异常的DDoS检测取决于**检测模型的建立**，不同的模型对应着不同的检测方式，主要包括**统计检测**、**模式预测**、**人工智能检测**、**机器学习检测**四种方法。

  - 统计检测的方法是用统计的方法计算出网络正常工作时流量的阈值，然后与当前网络流量进行比较，如果当前网络流量超过了阈值则说明可能发生了DDoS攻击。

  - 模式预测就是通过分析攻击发生前必然发生的一些现象来判断是否发生了DDoS攻击。

  - 人工智能检测方法主要包括数据挖掘、人工神经网络和模糊理论等。

  - 使用机器学习的方法实现DDoS攻击的检测也是可行的。

混合模式DDoS检测

- 混合模式DDoS攻击检测是将误用DDoS攻击检测和异常DDoS攻击检测**两种方式混合使用**。

- 通常使用数据挖掘的方法，由异常检测发现攻击，从发现的攻击中摘录特征放入误用模式特征库中，再利用误用检测的方法来检测DDoS攻击。但实际效果根据具体情况的不同也有差异。

按算法部署位置分类的检测方法

- **源端检测**

  - 源端DDoS攻击检测指的是将检测算法布置在发出攻击数据包的主机所处网络的边界路由器上。
  - 将DDoS攻击检测系统部署在源端，可以使得攻击数据流在进入网络之前被阻止。

- **中间网络检测**

  - 中间网络DDoS攻击检测是指将攻击检测算法部署在整个网络上，包括路由器、交换机或其他网络设备。
  - 在中间网络进行检测,通常是在核心路由器上部署分布式的DDoS防御检测系统。

- **目的端检测**

  - 目的端DDoS攻击检测是指将攻击检测算法部署在被攻击的主机和相关网络设备上。目前应用得最多的攻击检测都是在目的端(即受害端)进行的。

DDoS攻击检测方法的分析

表 1　DDoS 攻击检测技术比较

DDoS 检测技术	指标						
	检测率	误报率	漏报率	收敛时间	算法复杂度	建模难易	维护开销
特征匹配	相当高	相当低	高	实时	低	较易	较大
模型推理	较高	较低	较高	较快	较高	难	较大
状态转换	较低	较低	较高	快	较低	较易	较小
专家系统	较低	较低	较高	较慢	较高	难	较大
统计	较高	低	较低	较快	较低	较易	较小
模式预测	较高	较低	低	较慢	高	较难	较小
系统调用	较低	相当低	高	快	低	较易	较小
数据挖掘	较高	较低	较低	较慢	较高	较易	较大
神经网络	较高	较低	较低	较快	较低	较易	较小
模糊理论	较高	较低	较低	快	低	较难	较小

表 2　DDoS 攻击检测算法部署位置比较

部署位置	指标						
	检测率	误报率	漏报率	部署难易	路由器开销	正常报文存活率	管理开销
源端	较低	较低	较高	难	较小	大	较小
中间网络	较高	较低	较低	难	较小	较大	大
目的端	高	较高	低	易	大	小	较小

例：网站屏蔽

- 物理
  - 断电、断网
- 配置
  - IP、网关、DNS
- 链路
  - ARP欺骗
- 网络
  - IP阻断、TCP重置、数据替换等
- 应用
  - DNS欺骗

如何让指定用户访问不了北邮主页？

除传统的各类DoS之外，还有更多的方法实现。

问题和讨论