仅需要做深度分组检查，而且必须要将每个过往的分组与数以万计的“特征（signature）”进行比较；这可能导致极大的处理量，特别是如果机构从因特网接收每秒数十亿比特的流量时更是如此。将 IDS 传感器进一步向下游放置，每个传感器仅看到该机构流量的一部分，维护能够更容易。无论如何，目前有许多高性能 IDS 和 IPS 系统可供使用，许多机构实际上能够在靠近其接入路由器附近只使用一个传感器。

IDS 系统大致可分类为**基于特征的系统**（signature-based system）或**基于异常的系统**（anomaly-based system）。基于特征的 IDS 维护了一个范围广泛的攻击特征数据库。每个特征是与一个入侵活动相关联的规则集。一个特征可能只是有关单个分组的特性列表（例如源和目的端口号、协议类型和在分组载荷中的特定比特串），或者可能与一系列分组有关。这些特征通常由研究了已知攻击、技艺熟练的网络安全工程师生成。一个机构的网络管理员能够定制这些特征或者将其加进数据库中。

从运行上讲，基于特征的 IDS 嗅探每个通过它的分组，将每个嗅探的分组与数据库中的特征进行比较。如果某分组（或分组序列）与数据库中的一个特征相匹配，IDS 产生一个告警。该告警能够发送一个电子邮件报文给网络管理员，能够发送给网络管理系统，或只是做日志以供以后检查。

尽管基于特征的 IDS 系统部署广泛，但仍具有一些限制。更重要的是，它们要求根据以前的攻击知识来产生一个准确的特征。换言之，基于特征的 IDS 对不得不记录的新攻击完全缺乏判断力。另一个缺点是，即使与一个特征匹配，它也可能不是一个攻击的结果，因此产生了一个虚假告警。最后，因为每个分组必须与范围广泛的特征集合相比较，IDS 可能处于处理过载状态并因此难以检测出许多恶意分组。

当基于异常的 IDS 观察正常运行的流量时，它会生成一个流量概况文件。然后，它寻找统计上不寻常的分组流，例如，ICMP 分组不寻常的百分比，或端口扫描和 ping 掠过导致指数性突然增长。基于异常的 IDS 系统最大的特点是它们不依赖现有攻击的以前知识。在另一方面，区分正常流量和统计异常流量是一个极具挑战性的问题。迄今为止，大多数部署的 IDS 主要是基于特征的，尽管某些 IDS 包括了某些基于异常的特性。

Snort

Snort 是一种公共域开放源码的 IDS，现有部署达几十万［Snort 2012；Koziol 2003］。它能够运行在 Linux、UNIX 和 Windows 平台上。它使用了通用的嗅探接口 libpcap，Wireshark 和许多其他分组嗅探器也使用了 libpcap。它能够轻松地处理 100Mbps 的流量；安装在千兆比特/秒流量速率下工作，需要多个 Snort 传感器。

为了对 Snort 有一些认识，我们来看一个 Snort 特征的例子：

```
alert icmp $EXTERNAL_NET any -> $HOME_NET any
(msg:"ICMP PING NMAP"; dsize: 0; itype: 8;)
```

这个特征由从外部（$EXTERNAL_NET）进入机构网络（$HOME_NET）的任何 ICMP 分组所匹配，其类型是 8（ICMP ping）并且具有空负载（dsize = 0）。因为 nmap（参见 1.6 节）用这些特定的特征产生这些 ping 分组，所以设计出该特征来检测 nmap 的 ping 扫描。当某分组匹配该特征时，Snort 产生一个包括“ICMP PING NAMP”报文的告警。

也许关于 Snort 印象最为深刻的是巨大的用户社区和维护其特征数据库的安全专家。通常在一个新攻击出现后的几个小时内，Snort 社区就编写并发布一个攻击特征，然后它就能被分布在全世界的数十万 Snort 部署者下载。此外，使用 Snort 特征的语法，网络管理员能够根