第3章

系统日志数据采集

本章学习目标

- 了解系统日志数据概念、分类、采集方法等基础知识；
- 了解 Flume、Scibe 日志数据采集系统；
- 熟练掌握 EventLog Analyzer、Log Parser 日志采集系统。

日常工作中，机器每天会生成大量日志文件，日志文件一般由数据源系统生成，用于记录数据源执行的各种操作活动，如网络监控的流量管理、计算机登录退出信息以及访问 Web 服务器行为等。在大数据时代，能够从日志文件中得到众多有价值的数据。通过收集这些日志文件，然后进行数据分析，可以从公司业务平台的日志文件中挖掘潜在的价值信息，为公司的决策和后台服务器平台的性能评估提供可靠的信息保障。系统日志采集系统所做的就是采集不同类型的日志文件，并支持离线和在线的实时查找、分析、筛查，找出有用的资源。

目前，越来越多的企业建立日志采集系统来保存大量的日志数据，并通过对这些数据进行分析来获取商业或社会价值。流行的工具包括 Cloudera/Apache 的 Flume、Facebook 的 Scrible、Manage Engine 的 EventLog Analyzer 和微软的 Log Parser。

3.1 系统日志数据采集概述

通用日志采集系统由采集、存储分析和结果应用三部分组成。日志采集主要负责提供多种方式、方法进行采集日志；日志存储分析主要实现日志统一存储和定制的场景分析；结果应用实现将日志分析结果提供服务接口或者默认的管理功能，供各种应用功能使用。

3.1.1 系统日志分类

系统日志记录系统中的硬件、软件、系统问题等信息，以及根据不同命令执行的各种操作活动。根据系统日志记录的内容，系统日志一般分为三类：用户行为日志、业务变更日志和系统运行日志。

1. 用户行为日志

记录系统用户使用系统时的一系列操作信息，登录/退出的时间、访问的页面、使用的不同应用程序等。

2. 业务变更日志

该日志主要根据需要，用于特定业务场景。它记录了用户在某一时刻使用某一功能对某一业务（对象、数据）进行了何种操作，例如，何时从 A 变为 B 等信息。

3. 系统运行日志

该日志用于记录系统运行时，服务器资源、网络和基础中间件的实时运行状态，定期从不同设备采集信息进行记录。

以上这些种类的日志数据，记录了用户方方面面的行为，包括各种交易、社交、应用程序习惯、搜索等信息。通过对这些数据的收集，制定规则进行分析，对结果进行筛选，可以获得拥有各种商业价值和社会价值的资料。

3.1.2 日志分析应用场景

日志收集和分析是由需求驱动的。日志采集根据某一场景提出的需要，进行指定内容的采集，并对采集到的日志有针对性制定规则，使用专业软件手段进行分析。分析后的结果常见的应用场景包括如下几种。

1. 完善和提升系统

应用程序通过分析用户行为，调查哪些功能最受欢迎，对用户信息进行分析，得出某区域、某类用户为何种客户群，这有利于针对指定用户进行功能提升，有利于完善应用程序功能，从而提高用户体验。

2. 内容推荐

根据用户平时的阅读内容，收集相关日志并进行分析，通过完善算法，自动向用户推荐其感兴趣的内容，提高用户黏性。

3. 系统审计

对于企业（商业）公司为用户提供的系统，收集日常操作日志和业务变更日志有利于备查，也可以提供相关的安全审计功能。

4. 自动化操作和维护

场景化服务架构的系统或平台，在运维上需要很高的投入。自动化部署和运维，可以从很大程度上减轻运维的工作量。针对系统运行环境日志进行自动采集和分析，根据预定规则可以实现服务器资源的预警和动态分配，有利于快速定位和排除故障。

现在使用的系统多种多样，不同的系统有不同的运行环境，采集的日志类型大不相同，相应分析日志的需求也有较大差别。因此日志内容和采集方式具有多样性，有必要设计一个日志采集系统，满足日志采集的需要，并在应用功能中支持便捷反向捕获分析结果。

3.1.3　系统日志收集方法

微课视频

收集系统日志有三种方式。

1. Web API 模式

日志数据以基于 HTTP 的 RESTful 模式采集，并发送到消息队列，主要用于提供移动端、微信公众号和少量日志采集，可与.NET 分布式系统上的“API 网关”结合使用。

2. 服务代理模式

根据日志组件和消息队列客户端驱动程序，将其封装为日志服务代理，提供方便统一的使用界面。支持日志在本地和在线实时发送消息，其中日志发送到本地可以结合第三种方式完成日志采集的功能。

3. LC 客户端模式

实现客户端批量抓取日志数据，并将其发送到 LC 服务器。LC 客户端基于 TCP 与 LC 服务器通信，基于 NIO 框架构建，可以支持高并发处理。随后，LC 服务器将日志数据写入文件。

3.2　Flume 数据采集

Flume 最初是 Cloudera 公司提供的一个用于收集、聚合和传输海量日志的高可用、可靠、分布式的工具。Flume 支持在日志系统中定制各种数据发送规则，用来采集数据。同时，Flume 提供了简单处理数据并将其写入各种存储格式（可定制）的方法。

3.2.1　Flume 效益

目前 Flume 分为两个版本：①Flume0.9x 版本，统称为 Flume-OG（原代）；②Flume1.x 版本，统称为 Flume-NG（下一代）。由于早期的 Flume-OG 存在设计不合理、代码臃肿、扩展困难等问题，在 Flume 被接纳进 Apache 后，开发人员对 Cloudera Flume 的代码进行了重构，对 Flume 的功能进行了补充和加强，并将其更名为 Apache Flume，于是就有

了 Flume-NG 和 Flume-OG 两个完全不同的版本。在实际开发中，大多使用 Flume-NG 的流行版本进行 Flume 开发，本节主要对 Flume-NG 进行介绍。

利用 Flume 采集系统日志数据具有以下优点。

1. 集中存储

Flume 可以将应用程序生成的数据，存储在任何集中式存储器中，如 HDFS、HBase。

2. 高稳定性

当数据采集的速度超过需要写入的数据时，也就是当采集的信息满足峰值时，采集的信息非常大，甚至超过了系统写入数据的能力，此时 Flume 会在数据生产者和数据接收者之间进行调整，以确保其能够提供介于两者之间的稳定数据。

3. 传输和接收的一致性

Flume 的采集流水线是基于事务的，因此保证了数据传输和接收的一致性。

4. 整体性能高

Flume 具有可靠性、容错性、可扩展性、可管理性和可定制性。

3.2.2 Flume 整体结构

Flume 是一个分布式日志数据采集系统，它从各种服务器上采集数据并发送到指定的地方，如 HDFS、HBase 等，其总体结构见图 3-1。

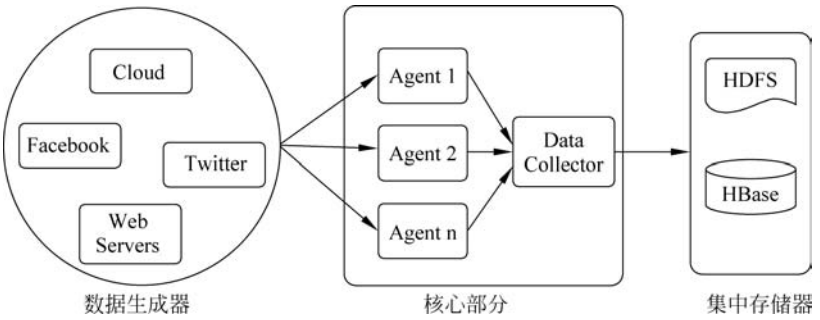

图 3-1 Flume 总体结构

从图 3-1 可以看出，一个基于 Flume 的日志采集系统由三部分组成：数据生成器、Flume 核心部分和集中存储器。其中，数据生成器（如 Facebook、Twitter 等）生成的数据，由运行在 Flume 服务器上的单个代理收集，然后数据收集器从每个代理收集数据，并将收集到的数据存储在 HDFS 或 HBase 中。

在数据传输的整个过程中，事件在流动。事件是 Flume 内数据传输的最基本单位。它由一个可选的报头和一个加载数据的字节数组（数据集从数据源接入点传入并传输到发射机）组成。

Flume 系统首先对事件进行封装，然后使用 Agent 对其进行解析，并根据规则将其传输到 HBase 或 HDFS 中。

3.2.3　Flume 的核心部件 Agent

Flume 的核心是 Agent。Flume 将 Agent 作为最小的独立运行单元，一个 Agent 就是一个 Java 虚拟机(JVM)。Agent 是一个完整的数据采集工具，包括三个核心组件，即数据源(源)、数据通道(通道)和数据槽(宿)。通过这些组件，“事件”(event)可以从一个地方流到另一个地方。各组件的具体功能见图 3-2。

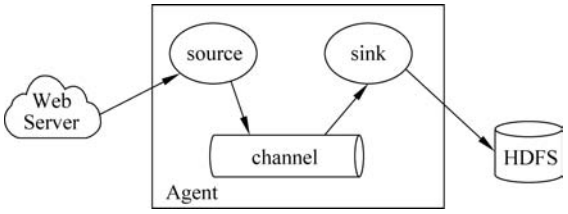

图 3-2　Flume 核心 Agent 组件

1. 数据源(source)

数据源是数据采集端，负责数据捕获后的特殊格式化，将数据封装成事件，再将事件推送到数据通道。Flume 支持的数据源有：

(1) Avro 源。Avro(数据序列化系统)源负责监控 Avro 端口，接收来自外部 Avro 客户端流的事件数据，与另一个 Flume Agent 上的 Avro sink 配对时，可以创建一个层次集合拓扑，使用 Avro 源可以实现多级流、扇出流、扇入流等效果。

(2) 假脱机目录源。Spooling(假脱机)目录源允许监视指定磁盘上的文件目录提取数据，它会查看文件指定目录中新添加的文件，并读出文件中的数据。

(3) Taildir source。Taildir(尾巴)source 用于观察指定的文件，可以实时地监视每个文件中添加的新行。如果文件正在写入新行，这个收集器将重新收集它们以等待写入完成。

(4) HTTP 源。HTTP 源可以通过 HTTP POST 和 GET 请求接收事件数据，GET 通常只用于测试，HTTP 请求会由 HTTP 源处理程序的 handler(处理器)可插拔插件接口实现到 Flume events 中，这个处理程序接收 HttpServletRequest 并返回 Flume events 列表。

2. 数据通道(channel)

数据通道是连接数据源和数据时隙的组件，可以看作是数据的缓冲区(数据队列)。它可以将事件临时存储在内存中，或者将其持久化到本地磁盘，直到数据槽完成对事件的处理。Flume 支撑的数据通道有：

1) 内存通道(memory channel)

内存通道将事件存储在可配置最大尺寸的内存队列中，非常适合需要较高吞吐量的

流量，但代理失败时会丢失部分数据。

2）文件通道(file channel)

文件通道是 Flume 的一个持久通道，它将所有事件写入磁盘文件，因此，不会因丢失进程或机器关闭，造成数据丢失。File 通过在一个事务中提交多个事件来提高吞吐量，这样只要提交事务，就不会丢失数据。

3. 数据槽(sink)

数据槽取出数据通道中的数据，存储到文件系统和数据库，或者提交到远程服务器。sink 类似一个集结的递进中心，它需要根据后续需求进行配置，从而最终选择是将数据直接进行集中式存储(例如，直接存储到 HDFS 中)，还是继续作为其他 Agent 的 source 进行传输。Flume 支持的 sink 有如下几种。

1）HDFS sink

HDFS sink 将 event 写入 Hadoop 分布式文件系统(HDFS)，它目前支持创建文本和序列文件，以及两种类型的压缩文件。HDFS sink 可以基于经过的时间或数据大小或 event 数量来周期性地滚动文件(关闭当前文件并创建新文件)，同时，它还通过属性(如 event 发生的时间戳或机器）来对数据进行分桶或分区。HDFS 目录路径可能包含将由 HDFS 接收器替换的格式化转义序列，以生成用于存储 event 的目录/文件名，使用 HDFS sink 时需要安装 Hadoop，以便 Flume 可以使用 Hadoop jar 与 HDFS 集群进行通信。

2）Logger sink

Loggersink 用于记录 INFO 级别 event，通常用于调试。Logger sink 接收器的不同之处是它不需要在“记录原始数据”部分中说明额外的配置。

3）Avro sink

Avrosink 形成了 Flume 分层收集支持的一半，发送到此接收器的 Flume event 将转换为 Avro event 并发送到配置的主机名/端口对上，event 将从配置的 channel 中批量获取配置的批处理大小。

3.2.4 Flume 运行机构

Flume 采用了分层架构，见图 3-3，分别为 Agent，Collector 和 Storage。其中，Agent 和 Collector 主要由 source 和 sink 两部分组成。

1. Agent

Agent 的作用是将数据源的数据发送给 Collector。Flume 自带了很多直接可用的数据源 Source。

2. Collector

Collector 的作用是将多个 Agent 的数据汇总后，加载到 Storage 中。它的 Source 和 Sink 与 Agent 类似。

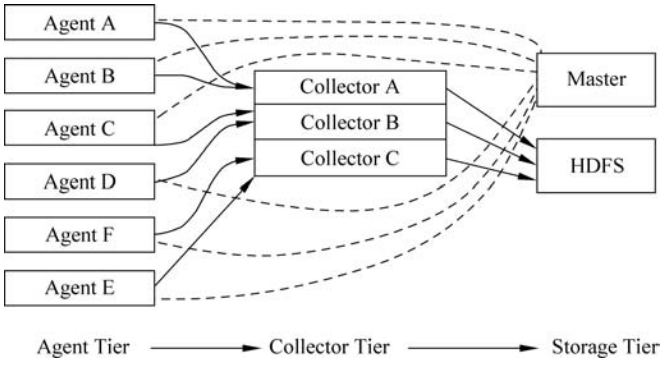

图 3-3　Flume 层次结构

3. Storage

Storage 是存储系统，可以是一个普通的 File，也可以是 HDFS、Hive、HBase、分布式存储器等。

4. Master

Master 负责管理、协调 Agent 和 Collector 的配置信息，是 Flume 集群的控制器。

为保证 Flume 的可靠性，Flume 在 source 和 channel 之间采用 Interceptors 拦截器用来更改或者检查 Flume 的 events 数据。在多 channel 情况下，Flume 可以采用默认管道选择器（每一个管道传递的都是相同的 events）或者多路复用通道选择器（依据每一个 event 的头部 header 的地址选择管道）实现 channel 的选择。为了保证负载均衡，采用 sink 线程用于激活被选择的 sinks 群中特定的 sink。

3.3　Scribe 数据采集

Scribe 是 Facebook 所发布的一个开源日志收集系统，已经得到了广泛的应用。Scribe 基于使用非阻塞 C++服务器的 thrift 服务的实现。它可以收集各种日志源的数据，并存储在集中存储系统（可以是 NFS 或分布式文件系统等）上，进行统一的统计、分析和处理。为日志的“分布式采集、统一处理”提供了一种可扩展、高度容错的方案。当集中存储系统的网络或服务器出现故障时，Scribe 会将日志转储到本地或另一台机器上。当集中存储系统恢复时，Scribe 会将转储的日志重新传输到集中存储系统。

Scribe 主要用于以下两种场景：一是它与生成日志文件的应用系统集成在一起。Scribe 提供了几乎所有开发语言的开发包，可以很好地与各种应用系统集成。二是应用系统本地生成日志文件。Scribe 使用一个独立的客户端程序（独立的客户端也可以用各种语言开发）来生成应用系统的本地日志文件。

3.3.1 Scribe 的功能

1. 支持多种存储类型

Scribe 支持多种存储类型，并且是可扩展的。

2. 日志自动分段功能

Scribe 可以自动按大小和时间拆分文件。

3. 灵活的客户

（1）支持多种通用语言。
（2）可与应用系统集成，作为独立客户端使用。

4. 支持日志分类

Facebook 有数百个博客类别。

5. 其他职能

（1）连接池。
（2）灵活的日志缓存大小。
（3）多线程功能（消息队列）。
（4）日志可以在划片服务器之间转发。

3.3.2 Scribe 的架构

Scribe 的架构比较简单，主要包括三个部分，分别是 Scribe Agent、Scribe 和存储系统，见图 3-4。

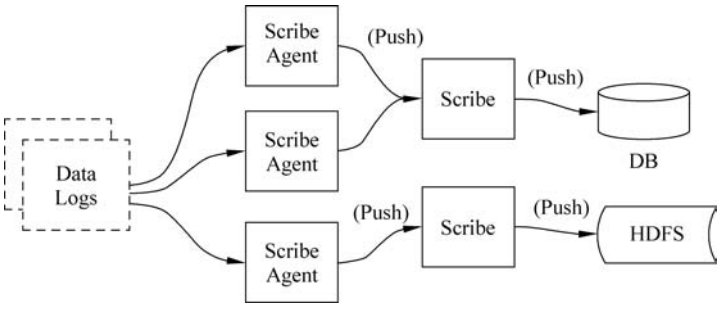

图 3-4 Scribe 体系结构

1. Scribe Agent

Scribe Agent 实际上是一个 thrift client。向 Scribe 发送数据的唯一方法是使用 Thrift Client，Scribe 内部定义了一个 Thrift 接口，用户使用该接口将数据发送给 server。

2. Scribe

Scribe 接收到 Thrift Client 发送过来的数据，根据配置文件，将不同 Topic 的数据发送给不同的对象。Scribe 提供了各种各样的 Storage，如 file、HDFS 等，Scribe 可将数据加载到这些 Storage 中。

3. 存储系统

存储系统实际上就是 Scribe 中的 Store，当前 Scribe 支持非常多的 Store，包括 File（文件）、Buffer（双层存储，一个主储存，一个副存储）、Network（另一个 Scribe 服务器）、Bucket（包含多个 Store，通过哈希将数据存到不同 Storage 中）、Null（忽略数据）、Thriftfile（写到一个 Thrift File Transport 文件中）和 Multi（把数据同时存放到不同 Store 中）。

3.3.3 Scribe 的流程

Scribe 采集数据的流程见图 3-5。Scribe 从各种数据源收集数据，将其放在共享队列中，然后将其推送到后端中央存储系统。当中央存储系统出现故障时，Scribe 可以临时将日志写入本地文件。在中央存储系统恢复性能后，Scribe 将继续把本地日志推送到中央存储系统。

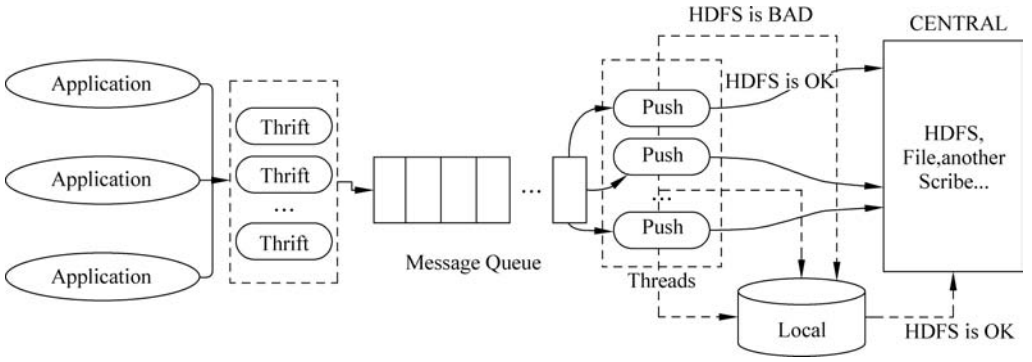

图 3-5　Scribe 数据采集流程

需要注意的是，每个数据源必须通过 Thrift 向 Scribe 传输数据（每个数据记录包含一个类别和一条消息）。可以在 Scribe 中配置用于侦听端口的 Thrift 线程的数量（默认值为 3）。在后端，Scribe 可以将不同类别的数据存储在不同的目录中进行单独处理。

3.3.4 Scribe 存储类型

1. file

Scribe 将日志写入文件或 NFS。支持两种文件格式，即 Std 和 HDFS，分别表示普通文本文件和 HDFS。

2. Network

Network 存储向其他 Scribe 服务器发送消息。Scribe 保持持久的链接打开以至于它能够发送消息。在正常运行的情况下，Scribe 会基于当前缓存中存在多少条消息等待发送而分批次的发送。

3. Scribe Buffer

Scribe Buffer 使用两层存储：一个主存储和一个辅助存储。日志将优先写入主存储，如果主存储失败，Scribe 会将日志临时存储在辅助存储中。主存储的性能恢复后，辅助存储中的数据将复制到主存储。

4. Null

用户可以在配置文件中配置一个名为 Default 的类别。如果数据所属的类别没有在配置文件中设置相应的存储方式，则该数据将被视为 Default。如果用户希望忽略此类数据，可以将其放入空存储区。

5. Bucket

Bucket 存储使用每个带前缀的消息作为 key 写入多个文件中。能够定义一个隐藏的或明确的 Bucket。定义隐藏的 Bucket 必须要有一个名称是“Bucket”子 Bucket，这个子 Bucket 可以是 File 存储、Network 存储或者 Thriftfile 存储。

6. Multi

一个 Mutil 存储是将所有消息转发到子存储中去的一个存储。一个 Mutil 存储可能有多个名为 Store0、Store1、Store2 等的子存储。

3.4 Event Log Analyzer 数据采集

Event Log Analyzer 是一个实时、基于 Web 的软件，主要用于日志收集和日常公司的事务管理和信息安全管理。Event Log Analyzer 支持多种日志类型，包括：不同系统（Windows、Linux、UNIX 等）的系统日志，网络设备（路由、交换机等），应用程序（数据库 Oracle、Apache 等）。该软件可对上述文件进行收集、查找、分析、报表、归档，提供了对网络中用户活动、网络异常和内部威胁的实时监测。IT 部门可用 Eveng Log Analyzer 进行网络系统审计，生成合规性报表。

3.4.1 Event Log Analyzer 特点

Event Log Analyzer 能分析所有 Windows 和 UNIX 系统日志。如果在网络中的某台机器上生成一个重要的安全事件，就会显示在 Event Log Analyzer 仪表盘上的即时报表中。从事件日志报表可以进行深入分析，即可在数分钟之内找出根本原因，然后集中力

量解决问题。

1. 即时告警

可以设置在服务器上生成特定事件时触发告警。例如，可以设置在邮件服务器上，将紧急事件通过电子邮件发送给操作员。借助 Event Log Analyzer 告警，就可以了解网络中每个系统的最新状态。

2. 将分布式事件存档到中央位置

存档的事件日志能充分显示系统在不同时间的性能。但是，事件日志检索是一项相当复杂的任务，除非将所有事件日志存储到一个中央位置，这样操作员就可以随时访问这些日志。Event Log Analyzer 能把从每个系统接收到的事件日志自动存档到一个中央位置，以供操作员随时访问。

3. 不需要客户端软件/代理

Event Log Analyzer 不需要在每台机器上安装单独的代理以便收集日志。因为收集 Windows 事件和 Syslog 消息的代理本身就是 Event Log Analyzer 服务器的一部分。因此 Event Log Analyzer 能在不增添主机负荷的前提下收集和分析事件日志。

然而，如果需要，可以将代理部署在客户端以便收集事件日志。这将便于在某个大型的分布式网络中，从各个位置的服务器收集事件日志。

3.4.2　Event Log Analyzer 主要功能

Event Log Analyzer 通过定义日志筛选规则和策略，帮助 IT 管理员从海量日志数据中精确查找关键有用的事件数据，准确定位网络故障并提前识别安全威胁，从而降低系统宕机时间、提升网络性能、保障企业网络安全。其主要功能如下。

1. 日志管理

日志管理是 Event Log Analyzer 的最主要功能，保障应用系统的网络安全，包括以下具体功能：

1）Windows 系统日志分析

监控和报表网络范围内的 Windows 服务器、系统和网络设备；以及合规问题、性能精度是一个重大责任。在如此重压的环境下，需要的将是一款前瞻性的事件日志监控解决方案。它能应付快速发展的 IT 世界，并提供高科技的、技术完备的 Windows 日志管理解决方案。

2）Syslog 分析

系统日志(Syslog)管理是几乎所有企业的重要需求。系统管理员将 Syslog 看作是解决网络上系统日志支持的系统和设备性能问题的关键资源。人们往往低估了对完整的 Syslog 监控解决方案的需求，导致长时间筛选大量系统日志来解决某一个问题。高效的事件日志 Syslog 分析可减少系统停机时间、提高网络性能，并有助于加强企业的安全

策略。

3）应用程序日志分析

Event Log Analyzer 支持广泛的应用程序，并使用其 ULPI 技术生成报表和告警。这些报表可分析生成的顶级事件、事件趋势等的详细信息。这些报表有助于 IT 安全经理轻松地管理拖欠费用户和分析应用程序的异常性能。

4）Windows 终端服务器日志监控

Event Log Analyzer 允许通过实时监控和分析终端服务器日志数据，监控 Windows 远程桌面服务上发生的用户活动并生成相关报表。它分析用户在每个远程连接会话中所花费的时间、登录到服务器的用户、所访问的资源、对用户授权的资源以及用户是否成功连接到资源等。

5）Syslog 服务器

Event Log Analyzer 带有内置的 Syslog 服务器。它通过侦听 Syslog 端口（UDP）来实时收集 Syslog 事件。另外，也可以配置多个端口来侦听 Syslog，当部分设备使用一些其他端口发送 Syslog 时，这很有用。在其他日志管理应用程序中，可能需要单独的 Syslog 服务器或转发器。

6）通用日志解析

Event Log Analyzer 比大多数设备和应用程序所能提供的丰富支持都更为强大，确保可以解析来自任何可生成人类可读日志的来源的日志数据。使用 Event Log Analyzer 可以分析网络中任何应用程序的日志数据。

2. 应用程序日志分析

Event Log Analyzer 的应用程序日志分析功能主要是对以下日志进行监控和分析。

1）IIS Web 服务器日志分析

Event Log Analyzer 收集、分析、审计和监控 IIS Web 服务器，并提供报表，包含错误事件、针对服务器的安全攻击和使用情况分析的信息。“错误”报表提供了对访问 IIS Web 服务器上托管的网站时可能遇到的错误（如用户身份验证失败、HTTP 错误请求、HTTP 请求实体太大和网关超时）的洞察，了解用户在网站上可能遇到的问题，且可向这些报表分配定制告警。

2）Apache Web 服务器日志分析

Event Log Analyzer 可以收集、分析 Apache HTTP 服务器生成的日志信息。使用这些信息，可以判断服务器的使用情况、找出漏洞所在，并设法改进服务器结构和整体性能。

3）打印机服务器日志分析

打印服务器是可能会导致公司信息泄露至关重要的地方，但是，确保打印服务器的顺利运行以及保持高效率的打印过程也很重要。监控打印服务器日志具备以下功能：确保不会打印敏感文档、标记试图在权限不足时进行打印的情况、识别失败的打印作业及其原因、发现员工打印习惯的模式和趋势等。

3. IT 合规性审计报表

Event Log Analyzer 的合规性审计报表功能主要是为了满足合规性审计需要。

1）合规性审计

Event Log Analyzer 通过实时监控网络和敏感数据，轻松生成合规报表，可帮助各组织为自定义时间段内的日志数据进行保留和归档。根据灵活的时间段归档日志数据可帮助管理员对归档日志执行取证分析，以满足合规审核要求、调查数据盗窃及追踪网络入侵者。

2）PCI 合规性报表

Event Log Analyzer 是 Manage Engine 的日志分析和管理软件，可帮助建立对 PCI DSS 的合规性。Event Log Analyzer 的合规报表主要满足 PCI DSS 第 10 条要求，该要求涉及跟踪和监控对持卡人数据的所有访问。除了第 10 条要求外，Event Log Analyzer 还可帮助确定是否符合其他 PCI DSS 要求。

3）ISO 27001 合规性报表

Event Log Analyzer 可以帮助企业生成符合 ISO 27001：2013 的 A.12.4.1、A.12.4.2 和 A.12.4.3 管制条例，这些管制条例可帮助企业记录事件并生成证据。同时，Event Log Analyzer 还满足管制条例 A.9.2.1、A.9.2.5 和 A.9.4.2 的要求，确保仅允许授权用户访问系统和服务，并阻止对系统和服务未经授权的访问。

4）自定义合规性报表

Event Log Analyzer 允许用户自定义和配置现有的预定义合规性报表，以满足与企业及行业相关的独特内部安全策略要求。可以通过添加或删除默认事件报表列表来修改现有合规报表。

4. 系统与用户监控日志报表

Event Log Analyzer 的系统与用户监控日志报表功能可以及时了解事件活动。

1）内建报表

Event Log Analyzer 包含超过 1000 种预定义报表。每当收集日志时都会立即生成报表，通过仅显示最重要的信息来减少组织的日志审核开销。事件日志报表帮助组织分析其网络并满足各种安全和合规要求。

2）自定义报表

Event Log Analyzer 提供了几个选项来自定义报表。根据要求，可以使用新的报表配置文件创建新的自定义报表。除了新的自定义报表之外，还可自定义现有的预建（预填充）报表以满足需求。通过使用这些自定义报表，由于报表已针对特定数据进行删减，因此管理员可以轻松进行日志分析。

3）活动目录日志报表

Event Log Analyzer 可监控 Active Directory 日志，且任何特定故障事故可实时进行跟踪。它可以立即向网络管理员发出告警，因此可很快地采取补救措施来避免网络故障。

4）历史事件趋势

Event Log Analyzer 中的趋势报表可分析一段时间内设备的性能。趋势监视有助于预测网络上 Windows 和 UNIX 设备以及 Cisco 交换机和路由器的使用情况与性能。Event Log Analyzer 中的趋势报表根据事件严重性和事件类型显示跨设备生成的事件。

3.4.3 Event Log Analyzer 的可视化用户界面

Event Log Analyzer 是用来分析和审计系统及事件日志的管理软件，能够对全网范围内的主机、服务器、网络设备、数据库以及各种应用服务系统等产生的日志，进行全面收集和细致分析，通过统一的控制台进行实时可视化的操作。

在正常信息管理中，会涉及多种设备，见图 3-6。监控本地客户端、服务器、网络基础设备等提供的信息非常重要。本地报表可帮助分析生成的重要事件、事件趋势等详细信息。这些报表有助于 IT 安全经理轻松地管理用户和应用程序的异常性能，减少解决问题所花费的时间。

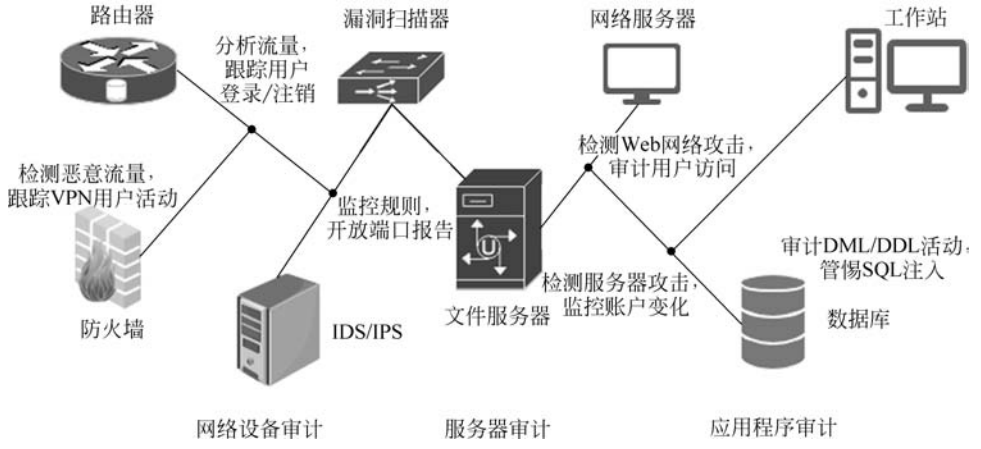

图 3-6 Event Log Analyzer 可监控设备

服务器是 IT 基础设施的关键，且事实上经常暴露于巨大的风险之下。人们通常认为服务器已经在防火墙的保护之下了，但实际上，它们很容易受到攻击，需要采取安全措施。主要考虑的因素是威胁源。随着远程连接技术的广泛使用，公司可以整合更多的远程工作者、全球办公室、企业以及竞争对手。远程员工可造成与内部员工相同的威胁，原因可能是不正确的安全措施以及对远程网络设备的监管不力。这就需要随时留意那些行为不当者、数据窃贼等。

除服务器和客户端外，监控网络设备是必需的，因为它可帮助用户全面了解网络。Event Log Analyzer 支持监视路由中的用户活动、路由流量、交换机日志、各种版本防火墙流量等。

从 Event Log Analyzer 的用户界面（见图 3-7），可以根据需要找到相应的功能。

用户界面从左到右依次如下。

1. Home(主页)

Home 选项卡包含集中的 Event Log Analyzer 设置面板,进一步分为多个选项卡,包括 Event Overview(本地机器)、Network Overview(网络事件)和 Security Overview(安全事件),可根据自身需要对主页面进行调整。主页面默认标签能看到基本选项,All Events 所有事件、Windows 事件、Syslog 事件、所有设备、日志趋势(Log Trend)等。

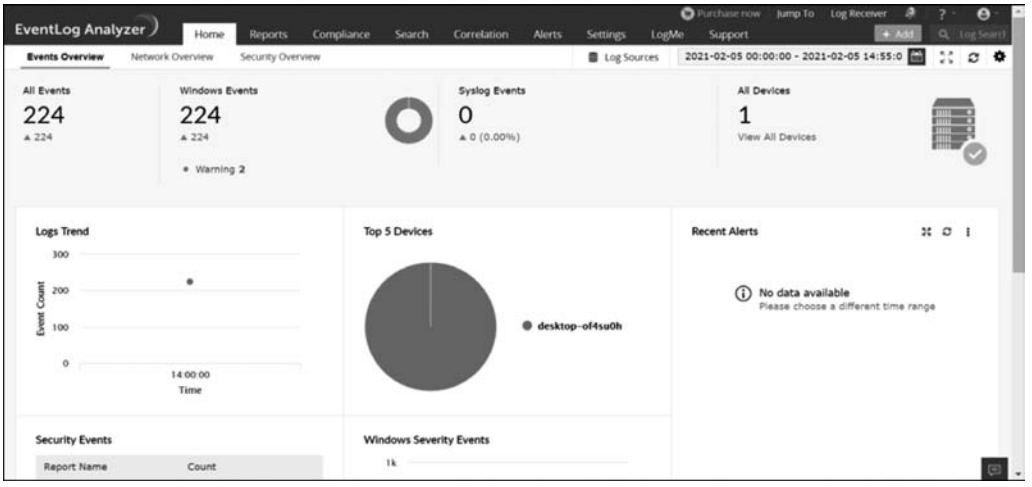

图 3-7　Event Log Analyzer 可视化界面

2. Reports(报表)

Reports 选项卡提供自定义报告和报告模板,可创建、修改、删除、计划和重新安排自定义报告。报告配置文件可以导入,并以 PDF、CSV 和 XML 格式导出。

报告模板见图 3-8,通常采用经常使用的信息自动生成报告,如用户活动报告、趋势报告、详细的应用程序报告和详细的设备报告。报告包括访问用户数量最多的设备、登录次数最多的用户、具有最多交互式登录的用户等。

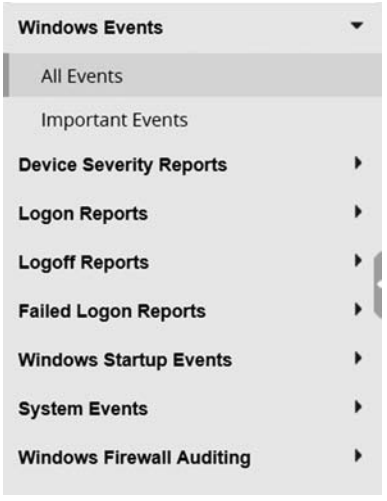

图 3-8　报告模板

3. Compliance(合规性)

Compliance 选项卡根据不同合规政策的要求生成指定的报告,如 FISMA(见图 3-9),包括 PCI-DS、SOX、HIPAA、GLBA、GPG 和 ISO 27001：2013 等。

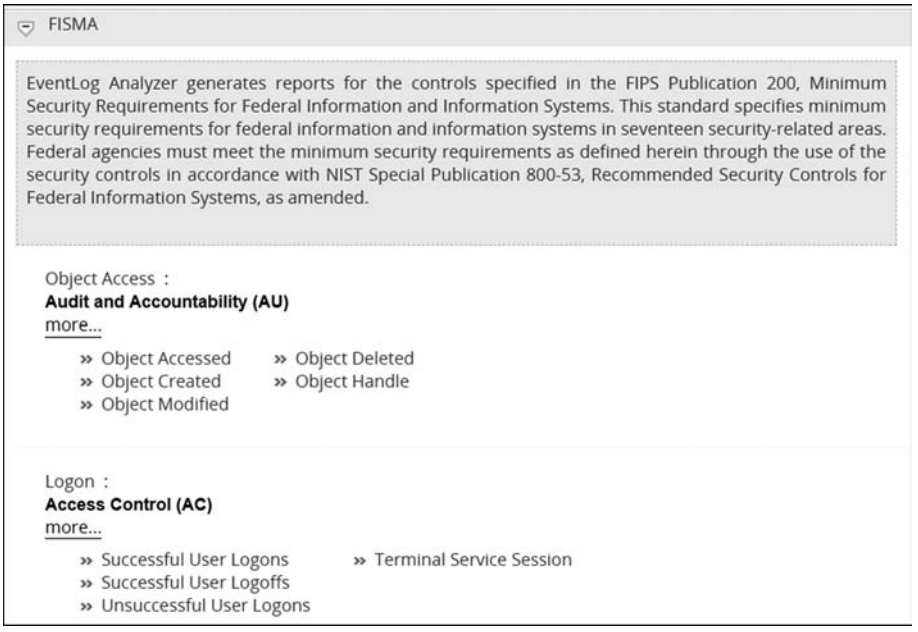

图 3-9　FISMA 报告

4. Search（搜索）

提供搜索原始日志的两个选项：基本搜索或高级搜索。搜索结果显示在页面的下半部分，最终搜索结果可以保存为报告（PDF 或 CSV 格式），也可以按预定间隔进行生成并自动发送邮件给指定用户。

图 3-10　不同规则预定义

5. Correlation（相关性）

提供不同类别之间的规则预定义，见图 3-10，这些规则会在发生特定事件（或事件组合），在指定时间段内进行指定次数时生成警报。比如制定网络相关规则，专门显示华为手机上网事件并定时将数据报表上传到指定负责人的邮箱。

6. Alerts（警告）

允许用户创建警报配置文件，以通知用户或团队阈值违规、网络异常、用户活动或合规违规等。警报选项卡显示所有警报配置文件、生成的警报，包括禁用、修改或删除任何现有警报配置文件的选项。警报配置文件也可以以 XML 格式导出或导入。

7. Setting(设置)

允许根据自身需求配置 Event Log Analyzer。三个子部分分别为“配置”“管理设置”和“系统设置”。

8. ＋Add(下拉菜单)

该按钮提供了添加主机、告警、报表和导入日志等功能。另外,搜索框用来进行快速搜索。在页面的右上角有两个图标,分别是 Event Log Analyzer 监听端口列表和 Syslog 数据包查看器。在帮助菜单中,提供了升级许可、关于、用户指南和反馈等选项。

3.5　基于 Log Parser 的数据采集

Log Parser 是一款功能强大的多功能工具,可提供对基于文本的数据(如日志文件、XML 文件和 CSV 文件)以及 Windows 操作系统上的关键数据源(如事件日志、注册表、文件系统和 ActiveDirectory)的查询以及输出。它可以像使用 SQL 语句一样查询分析这些数据,甚至可以把分析结果以各种图表的形式展现出来。

3.5.1　Log Parser 组成部分

Log Parser 主要组成部分,可以拆分为以下三种:目标输入、类 SQL 语句引擎、结果输出。通过架构图 3-11 可以看出,输入源(多种格式的日志源)经过 SQL 语句(有 SQL 引擎处理)处理后,输出想要的结果。

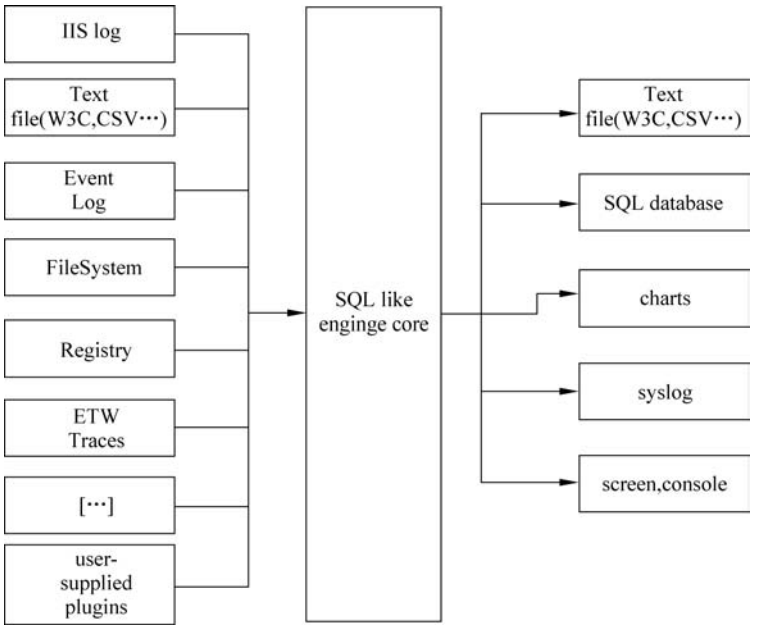

图 3-11　Log Parser 架构图

1. 目标输入

Log Parser 使用通用日志或记录当作输入来源的提供者，每条日志信息可以当作 SQL 表中的行，所以目标输入可以认为是要处理的数据的 SQL 表。

Log Parser 可以从以下数据类型中检索或处理指定数文件：

IIS 日志文件（W3C、IIS、NCSA、集中式二进制日志、HTTP 错误日志、URLScan 日志、ODBC 日志）、Windows 事件日志、Windows 注册表、通用 XML、CSV、TSV 和 W3C-格式化的文本文件（如 Exchange 跟踪日志文件、个人防火墙日志文件、Windows Media Services 日志文件、FTP 日志文件、SMTP 日志文件等）等。

2. 类 SQL 语句引擎

使用包含通用 SQL 语句（SELECT、WHERE、GROUP BY、HAVING、ORDER BY）、聚合函数（SUM、COUNT、AVG、MAX、MIN）和丰富的功能集（如 SUBSTR、CASE、COALESCE、REVERSEDNS 等）。

3. 结果输出

可以被视为接收数据处理结果的 SQL 表。Log Parser 的内置输出可以将结果以文本、图片存在本地，以数据形式发送上传。本地文本支持多种格式，如 CSV、TSV、XML、W3C 等。图片可以 GIF 或 JPG 格式导出，或直接显示，也可直接发送到 SQL 数据库、syslog 服务器等。

3.5.2 Log Parser Lizard 软件功能

针对 Log Parser 只能用类 SQL 编写，不易操作，引入了 Log Parser Lizard 软件。该软件将 Log Parser 进行了可视化处理，提高了可操作性，通过可视化操作，可以更直观地分析服务器日志、网站日志等复杂庞大的文件，方便导入和处理。Log Parser Lizard 同样支持多种格式的文件，如常用的日志文件、CSV 文件、XML 文件以及注册表、文件系统等内容。遇到 Windows 域被入侵的相关安全事件时，需要分析 Windows 安全日志，此类日志往往非常大，同样也可以使用 Log Parser Lizard 高效分析安全日志文件，提取有用信息，提高效率。

1. 封装了 Log Parser 命令

Log Parser Lizard 改进的可视化处理，增加了图形界面，减少了 Log Parser 对数据进行预处理时类 SQL 语句的输入，降低了 Log Parser 的使用难度。

2. 支持外观颜色设置

Log Parser Lizard 的软件主界面见图 3-12，可以通过不同颜色和图标区分不同数据。

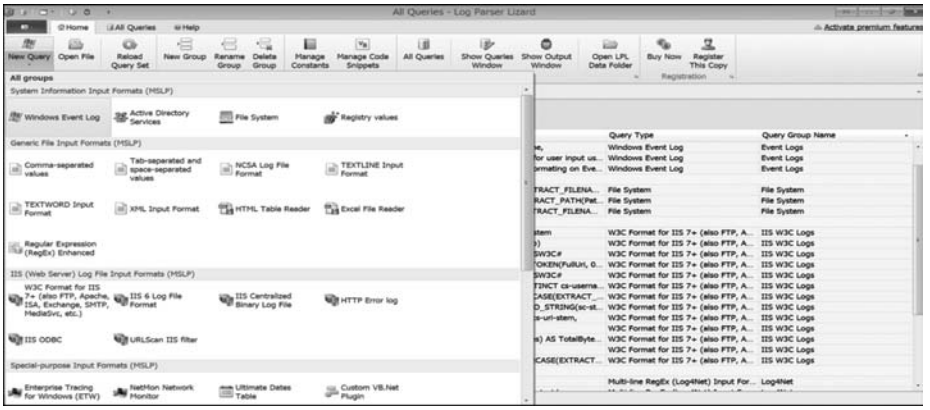

图 3-12　Log Parser Lizard 运行界面

3. 允许设置文件名

不同于 Log Parser 只能用命令行处理，Log Parser Lizard 可以针对不同内容从选项卡列表中新建文件名，删除和添加新的文件。

4. 允许保存文件

可以将编写过的命令，以文件的形式保存下来，方便下次直接打开调用。

5. 提供多种类图表

支持多种类的图表，见图 3-13，可以将查询结果以多种图表形式展示。

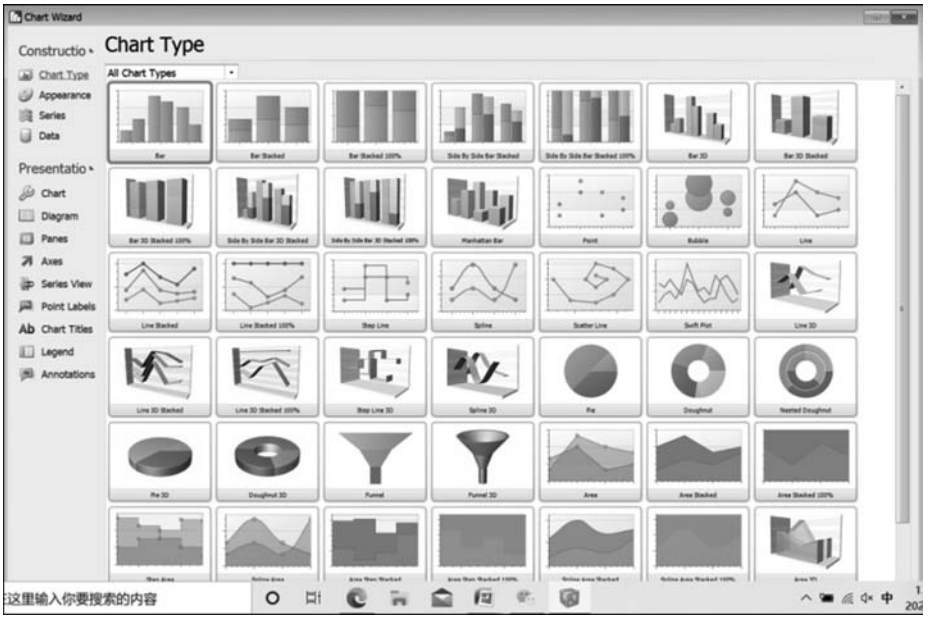

图 3-13　图表类型

6. 提供资源搜索功能

Log Parser Lizard 自带多种模板和数据库，可以直接引用。

7. 集成开源工具

如 log4net 等，可以方便地对 IIS logs、EventLogs、active directory、log4net、File Systems、T-SQL 进行查询。

8. 集成输出插件

集成 Infragistics. UltraChart. Core. v4. 3、Infragistics. Excel. v4. 3. dll 等，使查询结果可以方便地以图表或 Excel 格式展示。

9. 对纯文本文件运行 SQL 查询

通常，SQL 查询只能在数据库上执行，Log Parse 使用类似于 SQL 语法的语言。Log Parser Lizard 可以使用自身的 SQL 语法，能够对纯文本文件（如某些日志文件）直接进行 SQL 查询。例如，查找硬盘上最小的 10 个重复文件、相同状态码。Log Parser Lizard 也可以查找 IIS 日志、某个入侵的 IP 地址、HTTP 状态代码等。

10. 支持 Excel 和 PDF

运行所需的查询并查到所需的信息后，不仅可以保存为 CSV、IIS 日志文件，还可以将结果保存到 Excel 电子表格或 PDF 文件中，或直接将结果通过邮件附件发送出去。

3.5.3 Log Parser Lizard 软件特色

1. Expression Editor（表达式编辑器）

当表达式编辑器（见图 3-14）使用较复杂搜索条件时，可使用该模块加载多种布尔值或正则表达式。在编辑器中，可以手动输入表达式，或使用控件中提供的选择函数（Functions）、运算符（Operators）等。表达式是一个字符串，当进行解析与处理时，会计算一些值。表达式由列、字段名称、常量、运算符与函数组成。列、字段名称必须用括号括起来。

2. Conditional formatting for the column（过滤器）

该控件允许用户直观地使用可视化界面编写过滤器并编辑过滤条件，见图 3-15。

3. Result Grid（结果集控件）

该控件以表格形式显示结果集，与 Excel 表类似，支持对结果进行编辑、排序、筛选、分组、汇总计算等其他功能，见图 3-16。

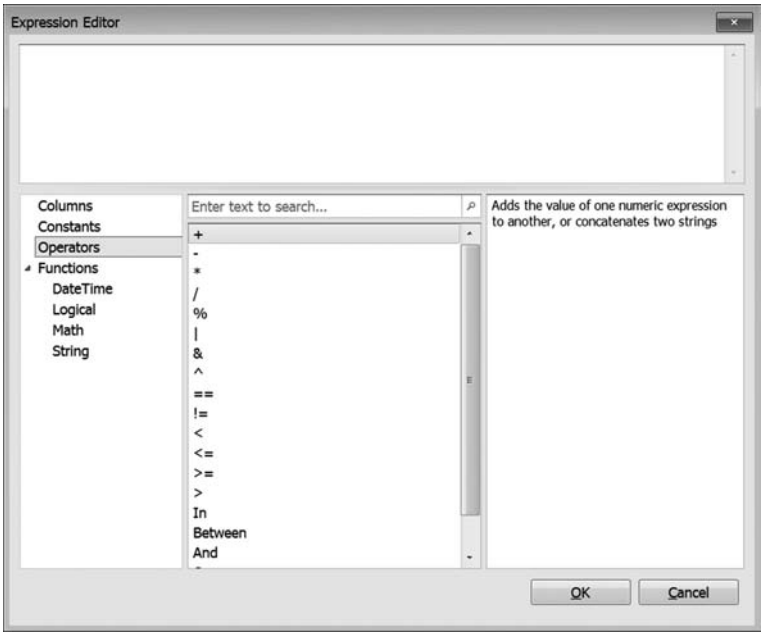

图 3-14　表达式编辑器

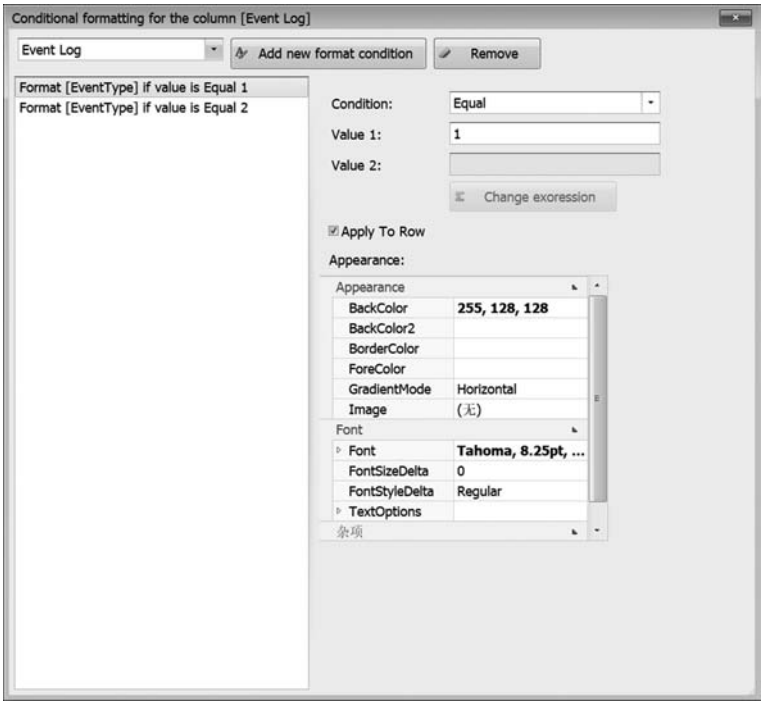

图 3-15　过滤器

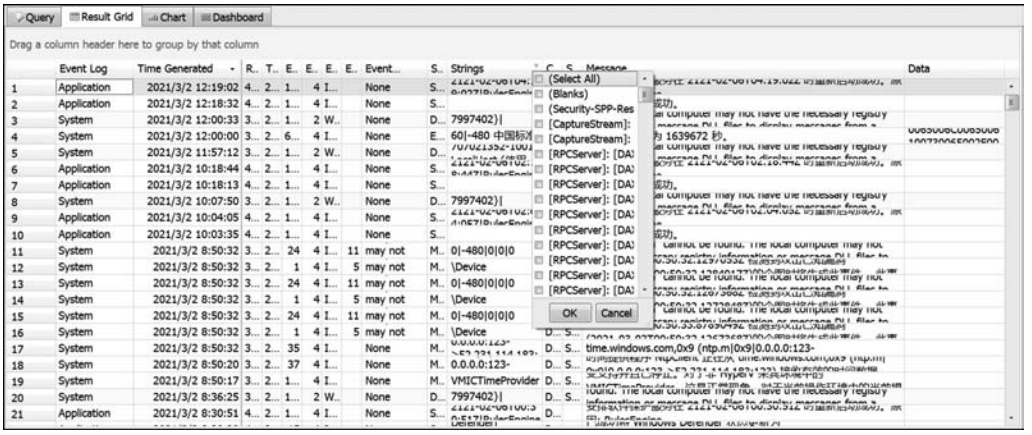

图 3-16　结果集控件

4. Dashboard（自制控制面板）

见图 3-17，Dashboard 模块可根据定制需要加载的类型，引用不同的数据源，制作包括数据透视表、图表、树形图等多种展示方式。

图 3-17　Dashboard

5. Report Designer（报告编辑器）

Report Designer 是一份 WYSIWYG 报告引擎，见图 3-18，该控件提供了一个用户界面，与 Word 操作界面类似，可以快速创建标准报告，具有数据整形（分组、排序与过滤）功能、基本文字排版、引入运行结果、分层数据表示等功能。

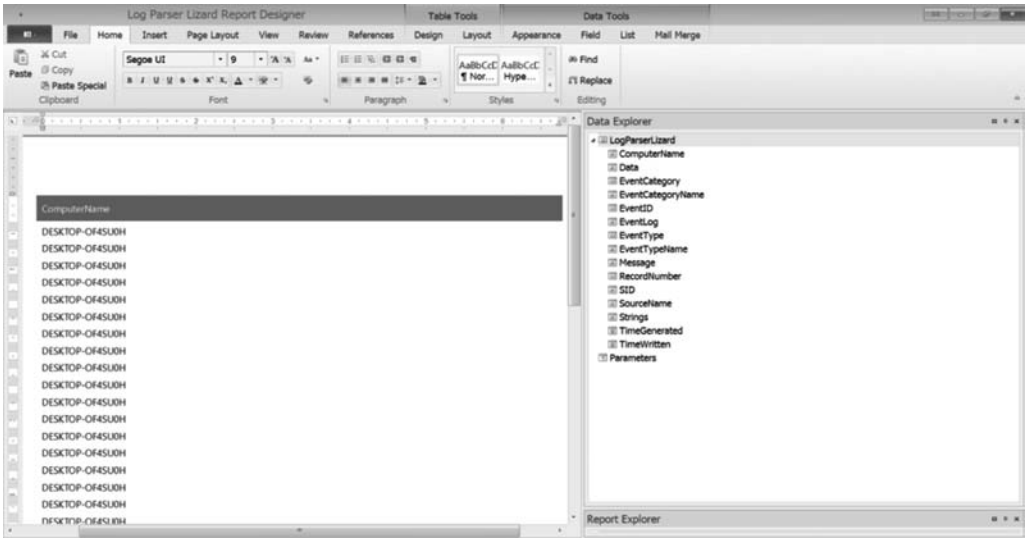

图 3-18　报告编辑器

3.6　本章小结

企事业单位的业务平台每天都会产生大量的日志文件，用于记录数据源执行的各种操作活动，包含很多有价值的数据。通过对这些日志文件进行采集，然后进行数据分析，可以挖掘到具有潜在价值的信息，为决策和后台服务器平台性能的评估提供可靠的数据保证。本章首先阐述了系统日志的分类、应用以及采集方法等基础知识，然后介绍了 Flume、Scribe、Event Log Analyzer 和 Log Parser 四种简单常用的采集工具。

习题

1. 系统日志分为哪些类别？
2. 系统日志的采集方式有哪些？
3. 利用 Flume 采集系统日志数据有什么优点？
4. Flume 的核心部件有哪些？
5. Scribe 架构的三个主要部分是什么？
6. Scribe 的存储方式有哪些？
7. Event Log Analyzer 应用于哪些场景？
8. Log Parser 由哪三部分组成？
9. Log Parser Lizard 的软件特色是什么？