# 基于Spark的实时攻击检测研究案例教学指导手册

**教学适用的课程：**《大数据架构与设计》

**教学适用的专业：**电子信息

**教学目的与用途：**熟悉Hadoop、Spark等大数据技术栈；掌握分布式大数据处理环境的搭建；掌握大数据处理与分析的基本流程。

**教学内容:**

（1）理论依据：本案例基于大数据技术对实时攻击进行建模和检测。案例整合了常用的大数据技术与深度学习方法，进行数据预处理与回归、检测，实现了分布式拒绝服务攻击的实时检测，从而能够更加快速、可靠的检测出入侵攻击,并及时作出应对。

（2）涉及知识点：流式实时处理；决策树回归；深度学习；数据可视化。

（3）分析路径：首先在Hadoop&Spark基础环境下，通过使用Flume+Kafka架构完成实时流式的日志处理，后面再连接上SparkStreaming流式实时处理技术，从而完成日志实时解析的目标。接着对数据集进行了分析和展示并介绍了相关字段含义。针对本次实验，分别用决策树与神经网络来进行模型训练，取二者准确率较高的一个作为最终模型。然后在当前场景下利用SparkStreaming进行实时检测，要点包括：1、在当前场景下对kafka的实时数据进行消费。2、将相应区间统计结果实时存入redis数据库中。3、将训练好的模型，从相应的存储位置取出，并广播到所有工作节点，避免工作节点重复拉取模型，造成性能问题。4、由于没有实际场景的web日志服务器，需要编写相应的脚本模拟日志的产生，将实时产生的日志数据添加到flume监控的日志文件尾端。最后利用matplotlib库对检测结果进行一个实时的展示。

**启发思考题:**

SparkStreaming流式处理框架与Storm流式处理框架的区别是什么？

（参考答案：1、Storm是纯实时的流式处理框架，SparkStreaming是准实时的处理框架（微批处理）。2、Storm 的事务机制要比SparkStreaming的要完善。3、Storm支持动态资源调度。4、SparkStreaming擅长复杂的业务处理，Storm不擅长复杂的业务处理，擅长简单的汇总型计算。）

**建议课堂计划：**

1. 时间安排：3课时
2. 学生学习准备：了解常见的大数据技术、决策树回归、深度学习、数据可视化的理论知识。
3. 分组及讨论内容：分组讨论启发思考题。
4. 案例开场白：网络入侵是现今异常入侵类型中最为普遍的一种入侵方式。计算机网络本身存在一些漏洞，再加之操作系统在早期忽略了安全问题，过于注重功能等原因，导致人们在使用网络时也不可避免的受到网络异常入侵的威胁。作为一种动态的网络安防手段，入侵检测技术因此应运而生，与静态安防技术相辅相成，共同构成网络安全防线。这节课我们将接触一个大数据背景下的基于Spark的网络入侵实时检测算法。
5. 结束总结：这节课我们基于Hadoop&Spark集群环境，通过Flume+Kafka架构和Spark Streaming流式实时处理技术，完成了日志实时解析的目标，接着通过引入决策树和深度学习预测模型，最终实现了入侵攻击的实时检测。我们通过平时课程所学知识去解决了实际社会生活中的问题，课后感兴趣的同学可以搜集相关资料，进一步了解大数据在其他领域的应用，相信你们会收获更多。
6. 案例引导建议：在教师简要分析应用场景后，鼓励学生自主考虑如何使用所学知识解决该场景问题。

**参考文献:**

[1] 周志华. 机器学习[M]. 北京: 清华大学出版社, 2016.

参考内容：数据预处理理论，线性回归、广义线性回归、决策树、SVM、logistic回归算法

[2] Goodfellow I, Bengio Y, Courville A. Deep learning[M]. MIT press, 2016.

参考内容：深度神经网络DNN模型

**其他教学支持材料：**

1. 本案例提供配套的PPT、视频、数据集与代码等，发布于Github，链接为：https://github.com/Wanghui-Huang/CQU\_bigdata。

2. 本案例涉及到数据预处理以及多种机器学习算法，建议使用python语言进行编写，推荐的工具包有pandas（数据读取与预处理库），scikit-learn（机器学习算法库），matplotlib（可视化绘图库）。