Sistema por processamento de fluxos em tempo real aplicado ao monitoramento da rede

Alexsander Haas[1], João V.F. Lima[1]

[1]Universidade Federal de Santa Maria (UFSM)
Caixa Postal 97105-900 – Santa Maria – RS – Brasil

{ahass,jvlima}@inf.ufsm.br

Resumo. *O presente trabalho apresenta um sistema para realizar o processamento de fluxos de dados em tempo real, baseado na arquitetura lambda. O sistema propõe-se a efetuar a análise sobre o tráfego da rede para obter informações que identifiquem acessos e processos indevidos, através da integração de ferramentas open source que executaram as etapas de geração dos dados, coleta, processamento e armazenamento.*

1. Introdução

Com a rápida evolução da internet ocorreu o aumento do volume, velocidade e variedade de dados que são gerados e transmitidos pela rede. Em conjunto com esta expansão, nos deparamos com o problema de como realizar um monitoramento efetivo da rede, para que seja possível realizar o processamento do seu fluxo de dados e obter os resultados de maneira mais rápida e executar ações sobre eles.

Com objetivo de apresentar soluções a esse problema novas arquiteturas foram propostas, conforme o conceito de arquitetura Lambda (LA) abordada por [Marz 2015] pois ela tem como objetivo unificar o processamento em lote (*batch layer)* que processa e armazena todos os dados, com o processo em tempo real (*speed layer*) responsável por integrar dados recentes de forma ágil e consumir novos dados, tendo os resultados de ambos processos exibidos (*serving layer*). Outro conceito de arquitetura que está sendo aplicado é a arquitetura Kappa (AK) abordada por [Kreps 2014], inspirado na LA, sendo uma alternativa de implementação simplificada, pois possui apenas duas camadas *speed* e *serving*, não efetuando o armazenamento de dados para realizar algum processamento em *batch*, dedicando-se a processar dados em tempo real.

Assim, considerando o acima exposto, este trabalho tem o objetivo de realizar a implementação de um sistema baseado na arquitetura lambda, aplicado para o processamento do fluxo de dados de rede com o intuito de identificar e monitorar acessos e utilizações indevidas.

2. Sistema proposto

A proposta de sistema visa permitir analisar o fluxo de dados em tempo real, efetuar o processamento de um grande conjunto de dados em que estão armazenados através de técnicas de computação distribuída e realizar a apresentação dos resultados agregando as informações obtidas nas duas camadas anteriores.

O sistema proposto bem como a sua estrutura lógica pode ser observado na Figura 1, que realiza a integração entre alguns *softwares open source*, tais como:

- **BRO (The Bro Network Security Monitor):** ferramenta responsável pelo monitoramento em tempo real do tráfego de rede, gerando logs de informações;
- **Apache Kafka:** serviço de mensagens distribuído, utilizado como um mediador de mensagens (*broker*) para abstrair todo o fluxo de dados em tópicos e por meio destes um terceiro consumir as informações;
- **Apache Spark:** utilizado para o processamento de dados em larga escala, de forma paralela e distribuída, aplicado ao processamento de fluxo de dados em tempo real;
- **Apache Phoenix:** é um mecanismo de banco de dados relacional, é utilizado para realizar o processamento de transações online, bem como sua análise. Possui integração com Spark e HBase.
- **Apache HBase:** base de dados distribuída, escalável e não relacional, executado no ecossistema do Hadoop.

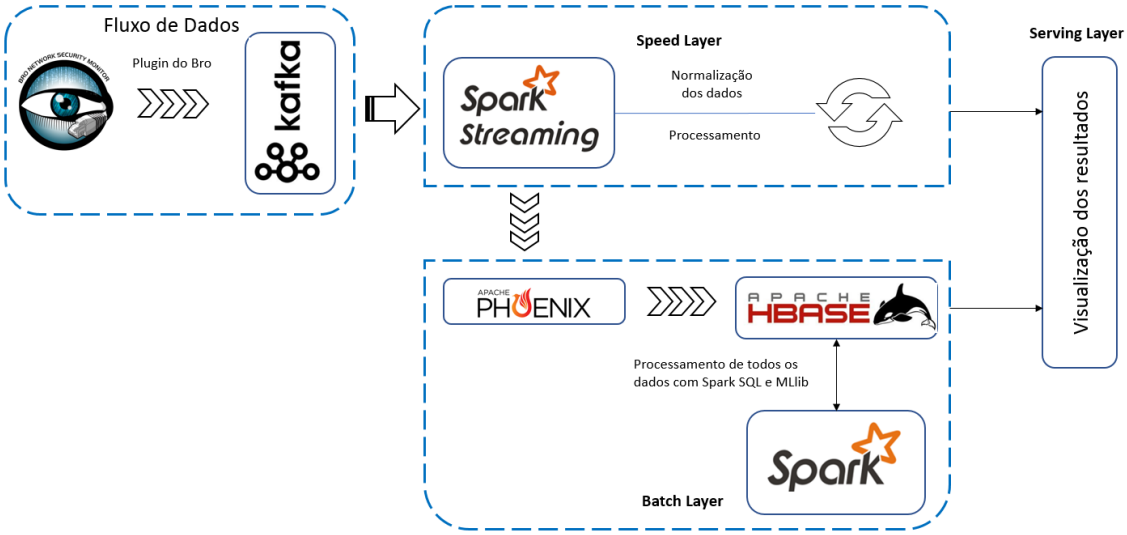

Figura 1 - Sistema proposto para processamento do fluxo de dados

O sistema possui o Bro como responsável por monitorar a rede e gerar o fluxo de informações com os dados do tráfego de rede, esse fluxo tem de ser transportado e para isso ocorre uma integração com o Kafka por meio do plugin do Bro, que atua como *producer* escrevendo todos os dados diretamente para os tópicos do Kafka. Tendo o fluxo de dados disponível o Spark *Streaming* é utilizado para atuar como *consumer* dos tópicos com isso é feito o processo para normalização dos dados, dessa forma eles ficam aptos para serem processados e, posteriormente, armazenados nas tabelas do HBase, utilizando como intermediário para essa operação entre Spark e HBase o Phoenix que realiza a escrita dos dados do *dataframe* para as tabelas.

3. Processamento realizado

Para o processamento inicial foi utilizado o conjunto de dados CICIDS2017 [UNB 2017] do dia 07/07/2017 com características de ataques DDoS e *Scan Port*. Como se trata de um conjunto de dados e não do fluxo de uma rede que está sendo monitorada, foi utilizado o BRO para realizar a leitura deste conjunto carregando todos os *scripts* implementados, com isso as informações obtidas deste arquivo são enviadas ao tópico Kafka e executa as demais etapas descritas na seção anterior.

Com os dados deste conjunto disponível para processamento em *batch*, foram selecionados os dados referentes a cada conexão e agrupados pelos campos destacados abaixo, realizando um *count* com o objetivo de identificar o período dos ataques DDoS:

- *Timestamp:* Por dia, hora e minuto
- Protocolo: TCP
- Serviço: HTTP
- IP de origem
- IP de destino
- Porta de destino

Com o resultado dos dados agrupados, o mesmo foi processado por meio do algoritmo de *cluster K-means*, para a classificação dos dados foi considerado o valor de *count* e a quantidade de classificações (*clusters)* utilizados foram dois.

4. Resultados Parciais

O sistema executou todas as etapas desde a leitura dos dados do arquivo até o processamento dos mesmos de forma precisa, o resultado obtido utilizando o algoritmo *K-means* mostra exatamente os mesmos períodos de ataques DDoS que foram informados na descrição do conjunto de dados, conforme pode ser observado no gráfico da Figura 2, o período do ataque é das 15:56 até 16:16.

Os centroides dos *clusters* definidos no processamento possuem os seguintes valores: 2.63 e 6705.08.

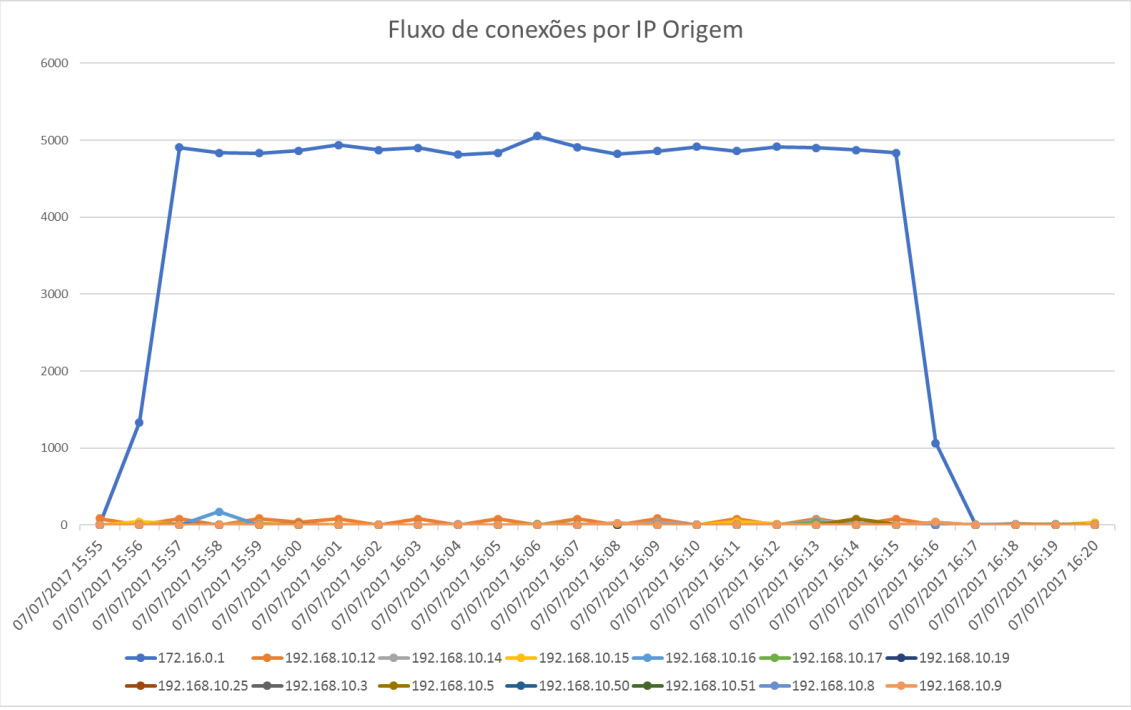

Figura 2 - Fluxo de conexões por IP Origem

Os IP's que apresentaram o maior número de conexões considerando o dia inteiro são apresentados no gráfico da Figura 3, tendo destaque o IP responsável por realizar o ataque DDoS com 95228 conexões, enquanto os demais não atingiram de cinco mil.

Figura 3 - Conexões por IP de todo o dia

5. Conclusões

Este trabalho traz a implementação de um sistema para realizar o processamento do fluxo de dados de rede em tempo real baseado na arquitetura lambda, apresentando assim uma opção de processamento de dados com baixa latência. Com a implementação realizada até o presente momento, é possível efetuar o monitoramento da rede por meio do BRO e realizar a transferência destes dados para serem normalizados e salvos por intermédio do Kafka, Spark, Phoenix e HBase.

Os resultados parciais apresentados até o momento são referentes ao processamento em *batch,* utilizando o algoritmo K-means para detecção de ataques DDoS, que foi efetivo sobre o conjunto de dados utilizado. A camada *streaming* no momento não está realizando processamento sobre os dados, mas realizando o processo de armazenar o fluxo de dados encaminhados por meio do Kafka.

Para trabalhos futuros será implementada a análise dos dados para detectar as varreduras de portas presentes no conjunto de dados utilizado no momento, utilizar os resultados que forem gerados na camada em *batch* referentes as análises de DDoS e *Scan Port* para processar os dados em *streaming*, e, posteriormente, aplicar o sistema para o monitoramento de uma rede em tempo real.

Referências

Kreps, J. (2014). Questioning the Lambda Architecture. https://www.oreilly.com/ideas/questioning-the-lambda-architecture. Acesso em: 05/08/2018.

Marz, N. and Warren, J. (2015). Big Data: Principles and Best Practices of Scalable Realtime Data Systems. Manning Publications, 1th edition.

University of New Brunswick (UNB). Intrusion Detection Evaluation Dataset (CICIDS2017). Disponível em: <https://bit.ly/2HcDIKh>. Acesso em: 16/11/2018.