Идеи:

* Анализ логов в распределенных системах. Поиск корреляций между логами различных компонентов.
* Анализ перфоманс тестов с использованием глубокого обучения. Прогнозирование времени следующего экшена моделью и сравнение реального времени. На основе этого делать вывод о изменении перфоманса.
* Анализ состояний систему на основе различных уровней мониторинга (pcap + logs analysis).
* Формулировка метрик нефункционального тестирования с точки зрения задачи машинного обучения
* Использование агрегаций по времени. Как входные фичи

### **DeepLog: Anomaly Detection and Diagnosis from System Logs through Deep Learning (deepLog.pdf)**

**ABSTRACT**

Anomaly detection is a critical step towards building a secure and trustworthy system. The primary purpose of a system log is to record system states and significant events at various critical points to help debug system failures and perform root cause analysis. Such log data is universally available in nearly all computer systems. Log data is an important and valuable resource for understanding system status and performance issues; therefore, the various system logs are naturally excellent source of information for online monitoring and anomaly detection. We propose DeepLog, a deep neural network model utilizing Long Short-Term Memory (LSTM), to model a system log as a natural language sequence. This allows DeepLog to automatically learn log patterns from normal execution, and detect anomalies when log patterns deviate from the model trained from log data under normal execution. In addition, we demonstrate how to incrementally update the DeepLog model in an online fashion so that it can adapt to new log patterns over time. Furthermore, DeepLog constructs workflows from the underlying system log so that once an anomaly is detected, users can diagnose the detected anomaly and perform root cause analysis effectively. Extensive experimental evaluations over large log data have shown that DeepLog has outperformed other existing log-based anomaly detection methods based on traditional data mining methodologies.

**KEYWORDS**

Anomaly detection; deep learning; log data analysis.

**SUMMARY**

Основная идея в том чтобы парсить лог как ключ + вектор значений. Обучаются 2  LSTM сети. Одна предсказывает следующий ключ, если он отличается от текущего ключа, то считается как аномалия, если ОК, то вторая сеть предсказывает следующее значение в векторе с каким-то запасом точности, если предсказанное значение выходит за доверительный интервал то детектится аномалия. Есть возможность дообучать веса на ложно положительных срабатываниях. Так же обсуждается анализ конкурент логов, когда в один файл может писаться логи нескольких процессов. Соответственно воркфлоу разных процессов могут пересекаться и их надо отделять друг от друга.

### **Anomaly Detection in Log Data using Graph Databases and Machine Learning to Defend Advanced Persistent Threats (1802.00259.pdf)**

**Abstract:** Advanced Persistent Threats (APTs) are a main impendence in cyber security of computer networks. In 2015, a successful breach remains undetected 146 days on average, reported by [Fi16]. With our work we demonstrate a feasible and fast way to analyse real world log data to detect breaches or breach attempts. By adapting well-known kill chain mechanisms and a combine of a time series database and an abstracted graph approach, it is possible to create flexible attack profiles. Using this approach, it can be demonstrated that the graph analysis successfully detects simulated attacks by analysing the log data of a simulated computer network. Considering another source for log data, the framework is capable to deliver sufficient performance for analysing real-world data in short time. By using the computing power of the graph database it is possible to identify the attacker and furthermore it is feasible to detect other affected system components. We believe to significantly reduce the detection time of breaches with this approach and react fast to new attack vectors.

**Keywords:** Advanсed Persistent Threat; Graph Database; Intrusion Detection; Machine Learning; Support Vector Machines; Kill Chain

**SUMMARY**

Говорится о наблюдении подозрительной активности юзеров. Используется БД для графов + SVM. В целом про анализ с точки зрение мл говорится не много. Главная идея юзать SVM.

### **Recurrent Neural Network Attention Mechanisms for Interpretable System Log Anomaly Detection (1803.04967.pdf)**

**ABSTRACT**

Deep learning has recently demonstrated state-of-the art performance on key tasks related to the maintenance of computer systems, such as intrusion detection, denial of service attack detection, hardware and software system failures, and malware detection. In these contexts, model interpretability is vital for administrator and analyst to trust and act on the automated analysis of machine learning models. Deep learning methods have been criticized as black box oracles which allow limited insight into decision factors. In this work we seek to “bridge the gap” between the impressive performance of deep learning models and the need for interpretable model introspection. To this end we present recurrent neural network (RNN) language models augmented with attention for anomaly detection in system logs. Our methods are generally applicable to any computer system and logging source. By incorporating attention variants into our RNN language models we create opportunities for model introspection and analysis without sacrificing state-of-the art performance. We demonstrate model performance and illustrate model interpretability on an intrusion detection task using the Los Alamos National Laboratory (LANL) cyber security dataset, reporting upward of 0.99 area under the receiver operator characteristic curve despite being trained only on a single day’s worth of data.

**KEYWORDS**   
Anomaly detection, Attention, Recurrent Neural Networks, Interpretable Machine Learning, Online Training, System Log Analysis

**SUMMARY**

Извлечение знаний из логов промышленных систем осложняется несколькими фактами: большой объем, разметить данные часто сложно и они несбалансированны или система-зависимы, взяли в данных могут быть скрыты за сложными отношениями источников логов и системных сущностей.

В этой работе используется RNN модель языка для анализа аномалий.  
Вклад: 1) оценка эффективности усиленной RNN модели языка несколькими механизмами внимания, 2) иллюстрация инстроспекции поведения системы.

Фишка работы в том что анализируется сырой текст логов, без предварительной обработки (токенизации и делимитеров). Далее авторы предлагаю способы интерпретации результатов предсказания аномалии, пытаются понять почему сеть сделала именно такой вывод.

Here we describe the unsupervised language modeling framework and its extension via five variations of attention. In each case, the language models consume a sequence of log-line tokens and output log-line-level anomaly scores.

**Recurrent Neural Network Language Models for**

**Open Vocabulary Event-Level Cyber Anomaly Detection (rec\_net\_lang\_model.pdf)**

Abstract

Automated analysis methods are crucial aids for monitoring and defending a network to protect the sensitive or confidential data it hosts. This work introduces a flexible, powerful, and unsupervised approach to detecting anomalous behavior in computer and network logs; one that largely eliminates domain-dependent feature engineering employed by existing methods. By treating system logs as threads of interleaved “sentences” (event log lines) to train online unsupervised neural network language models, our approach provides an adaptive model of normal network behavior. We compare the effectiveness of both standard and bidirectional recurrent neural network language models at detecting malicious activity within network log data. Extending these models, we introduce a tiered recurrent architecture, which provides context by modeling sequences of users’ actions over time. Compared to Isolation Forest and Principal Components Analysis, two popular anomaly detection algorithms, we observe superior performance on the Los Alamos National Labora-tory Cyber Security dataset. For log-line-level red team detection, our best performing character-based model provides test set area under the receiver operator characteristic curve of 0.98, demonstrating the strong fine-grained anomaly detection performance of this approach on open vocabulary logging sources.

SUMMARY

Используется 2 RNN сети. Одна для предсказания следующего символа в строке лога. Другая для предсказания следующего ивента в лог файле.

**Spell: Streaming Parsing of System Event Logs (spell.pdf)**

Abstract

System event logs have been frequently used as avaluable resource in data-driven approaches to enhance system health and stability. A typical procedure in system log analytics is to first parse unstructured logs, and then apply data analysis on the resulting structured data. Previous work on parsing system event logs focused on offline, batch processing of raw log files. But increasingly, applications demand online monitoring and processing. We propose an online streaming method Spell, which utilizes a longest common subsequence based approach, to parse system event logs. We show how to dynamically extract log patterns from incoming logs and how to maintain a set of discovered message types in streaming fashion. Evaluation results on large real system logs demonstrate that even compared with the offline alternatives, Spell shows its superiority in terms of both efficiency and effectiveness.

SUMMARY

Предлагается стриминговые алгоритм основанной на LS который парсит логи в вид сообщение + параметры

**Detecting Large-Scale System Problems by Mining Console Logs**

**Abstract**

Surprisingly, console logs rarely help operators detect problems in large-scale datacenter services, for they often consist of the voluminous intermixing of messages from many software components written by independent developers. We propose a general methodology to mine this rich source of information to automatically detect system runtime problems. We first parse console logs by combining source code analysis with information retrieval to create composite features. We then analyze these features using machine learning to detect operational problems. We show that our method enables analyses that are impossible with previous methods because of its superior ability to create sophisticated features. We also show how to distill the results of our analysis to an operator-friendly one-page decision tree showing the critical messages associated with the detected problems. We validate our approach using the Darkstar online game server and the Hadoop File System, where we detect nu- merous real problems with high accuracy and few false positives. In the Hadoop case, we are able to analyze 24 million lines of console logs in 3 minutes. Our methodology works on textual console logs of any size and requires no changes to the service software, no human input, and no knowledge of the software’s internals.

**SUMMARY**

* Анализ состояний систему на основе различных уровней мониторинга (pcap + logs analysis).

**-** Формулировка метрик нефункционального тестирования с точки зрения задачи машинного обучения

**Problem:**

Хочется получить информацию о состоянии системы за определенный период времени. Как это сделать? Проанализировать логи системы. По этому поводу написано большое количество работ: как распарсить, структурировать, извлечь features, натравить алгоритм машинного обучения (supervised или unsupervised), убедиться что прогноз аномалий адекватный и в конечном счете как-то визуализировать и интерпретировать результат.  
 Такой подход может хорошо сработать для систем которые прошли тщательное тестирование (а может не сработать) и можно предположить что система логирования работает корректно (опять же в ней могут быть ошибки). Поэтому единственный (не очень надежный) источник данных для постоения систем поиска аномалий с использованием машинного обучения не является хорошей идеей (особенно для критически важный систем). И уж тем более такой подход не стоит так делать для систем которые находятся на этапе тестирования.

**Solution:**

Использовать в качетсве подкрепления данные для анализа из нескольких источников. Например анализ логов + анализ pcap файлов (+ какие-то метрики системы, память, процессор). Применение такого комплексного подхода позволяет избежать использование единственного источника информации о состоянии системы. А так же найти дефекты в механизмах мониторинга и логирования самой системы.

**Our contribution:**

Our contribution is a general methodology that allows machine learning and information retrieval techniques to be applied using several undependent data sources for analysis the state of the under testing system.