OpenSOC Scalability
LIA - Project Proposal

Joey Dreijer       Carlo Rengo
Joey.Dreijer@os3.nl  carlo.rengo@os3.nl

February 20, 2015

Introduction

OpenSOC is an open source framework for real-time security analysis and data-analytics. The OpenSOC framework integrates a great part of the Apache stack (Hadoop, Kibana, Elastic search etc) to store, index and enrich data sources. These data-sources originate from network traffic and external log data. The goal of OpenSOC is to provide a true open source and scalable security analysis tool and provide, in many cases, an alternative to the expensive commercial SIEM-frameworks.

Research Questions

The main goal of our research is to test the scalability of the individual OpenSOC components. An OpenSOC/Apache stack will be configured and tested with simulated network flows and log streams. OpenSOC states on their Wiki: ”OpenSOC is designed to scale up to consume millions of messages per second, enrich them, run them through anomaly detection algorithms, and issue real-time alerts.” This research will focus on the following questions:

1. What are the scalability characteristics of OpenSOC in large-scale environments?
   a) What OpenSOC components will meet limitations when the amount of Events Per Seconds will increase?
   b) Can the individual components be scaled up/down without adjusting the overall infrastructure? (In the case of increased events)

After consideration, a few research questions are discarded from the main research. Due to the limited documentation of OpenSOC (i.e. almost none publicly available) it is difficult to perform tests on the OpenSOC correlation and enrichment/alerting engine. This research will (in the first place) focus on the OpenSOC Apache dependencies, since these are well-documented. The research will relate to the scalability of the platform rather than the possibility of detecting security events.
Requirements

Basing on the requirements of each single OpenSOC dependency, a proper hardware
and network setup is necessary to build a testing environment. Without giving a precise
description of the servers, networks, etc... we can safely assume that a good setup for
tests and benchmarking would include some nodes (virtualization will be used if/as much
as possible), a reasonable amount of RAM (i.e. 64GiB on overall) and at least 10TB of
storage.

Planning

<table>
<thead>
<tr>
<th>Week</th>
<th>Topic</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Reading of OpenSOC dependencies documentation</td>
</tr>
<tr>
<td></td>
<td>Setup of hardware/network infrastructure</td>
</tr>
<tr>
<td></td>
<td>Installation of OpenSOC dependencies</td>
</tr>
<tr>
<td>2</td>
<td>OpenSOC installation</td>
</tr>
<tr>
<td></td>
<td>Definition of test cases</td>
</tr>
<tr>
<td>3</td>
<td>Tests run and analysis of OpenSOC and each relevant dependency</td>
</tr>
<tr>
<td>4</td>
<td>Preparation of the Research Paper and presentation</td>
</tr>
</tbody>
</table>

Previous Work

There is little or no documentation about OpenSOC, as it is a “brand new” product (it
was released in September 2014, in beta state). The only documents available online are
some presentation made by Cisco during the last year but, although interesting, they are
more focused about describing the characteristics, the architecture and the challenges
faced during the development of OpenSOC. Since the product is still in beta, we guess
that this is the main reason why Cisco hasn’t published any benchmark yet.

However, since OpenSOC heavily relies on well known open source products, there
are a significant amount of works (sometimes even papers) describing their performance
and/or scalability individually. These researches will be therefore taken into account
with the ultimate goal of showing the scalability of OpenSOC as a single entity.

References


Cheap Machines)*. URL: https://engineering.linkedin.com/kafka/benchmarking-

