Proceedings of the 15th International Conference on Quality and Dependability
Sinaia, Romania, September 14th-16th, 2016
ISSN 1842-3566
Pages 176-180

A Reliable Architecture for a Massive and Continuous Scanner of Web Vulnerabilities in Internet

Eugenie STĂICUŢ, Radu BONCEA, Carmen ROTUNĂ
Romania Top Level Domain, National Institute for Research and Development in Informatics - ICI Bucharest
estaicut@rotld.ro, radu@rotld.ro, carmen.rotuna@rotld.ro

Abstract
In recent years, the Web has become one of the major vectors for transmitting malware and computer viruses. As a response, nations around the world have established Computer Emergency Response Teams with the purpose of countering the next generation of cyber threats. One such solution is for CERTs to pro-actively scan the Web for vulnerabilities and notify the right persons before malicious users could exploit the vulnerable application. Another solution is to search the Web for compromised and vulnerable applications and take appropriate actions, such as sending simple notifications to application's owner. Either way, continuously scanning of the Web is a complex task which requires a reliable architecture. In this paper we propose a data-centric architecture, with focus on a distributed streaming processing system. We will define a virtual process bus as a group of data channels where a process can take its input from a specific channel and write the result to an output set of channels.

Keywords: cybersecurity, stream processing, distributed processing, messaging, ETL, Kafka, vulnerability

References:

[1] Huseyin, Birendra Mishra, and Srinivasan Raghunathan. "The effect of internet security breach announcements on market value: Capital market reactions for breached firms and internet security developers", International Journal of Electronic Commerce 9.1 (2004): 70-104.
[2] Powers, J., Anderson, R., Trueblood, N., & Ciruli, D. (2005). U.S. Patent Application No. 11/245,952.
[3] https://access.redhat.com/documentation/en-US/Fuse_Message_Broker/5.3/html/Getting_Started/files/Fuse MBStartedKeyJMS.html
[4] http://www.jonathanbeard.io/blog/2015/09/19/streaming-anddataflow. html
[5] https://www.datadoghq.com/blog/monitoring-kafkaperformance- metrics/
[6] https://owasp.org
[7] https://dzone.com/articles/kafka-logs-and-the-policy-of-truth
[8] https://kafka.apache.org
[9] http://tools.kali.org/