**Discussion Topics: Telemetry and Anomalies**

Select one of the following and find at least one article on the topic.

1. Describe what is meant by Telemetry in DevOps. What type of data might be collected? Why?
2. What can be explained by a non-Gaussian distribution? Can the distribution still be used?
3. Provide at least two (2) examples of anomaly detections tools as they apply to DevOps. Which would you recommend? Why?

Within software development, detecting anomalies is important. When an anomaly is detected, it can improve the overall security of a system. Deployments can then be made if further security is required, otherwise it should handle these anomalies. Monitoring and tracking them should be standard conduct. Implementing an anomaly detection tool makes catching anomalies easier and an automated process. When bad actors attack, having a detection tool can alert developers before it is too late. There are many different software applications available to help detect anomalies. Software that detects anomalies looks for any events or input peculiar to the user's expected actions (PAT Research, 2020). Supervised detection involves looking for abnormal behavior for a given task based on a training dataset (PAT Research, 2020). Detection software finds anomalies in an unlabeled dataset referencing common baseline entries (PAT Research, 2020). Tracked anomalies can be sorted into single, collective, and contextual categories so that developers can better detect attacks (PAT Research, 2020).

PAT Research (2020) provides ten different examples of anomaly detection tools. The ones I will focus on are Numenta and Loom Systems. Numenta offers open-source technology for tracking anomalies in "servers and applications, human behavior, geo-spatial tracking data, prediction and classification of natural language" (PAT Research, 2020). Some anomalies detected include models' stock price, stock volumes, and other top markets (PAT Research, 2020). It also continues to learn, and paid commercial plans are available (PAT Research, 2020).

Loom Systems analyzes logs and metrics to understand expected user behavior and compare anomalies with (PAT Research, 2020). It is fully automated, with no configuration or log parsing required. It also provides real-time detection of issues and a complete analysis (PAT Research, 2020). Overall, Loom is the way to go if you want real-time issues and easy implementation.

**Reference**

PAT Research. (2020, June 19). *Top 10 Anomaly Detection Software in 2020 - Reviews, Features, Pricing, Comparison*. PAT RESEARCH: B2B Reviews, Buying Guides & Best Practices. https://www.predictiveanalyticstoday.com/top-anomaly-detection-software/

***Before you submit your thread, put your name in the subject line.***

**Assignment Requirements and Grading:**

1. An initial post of approximately 250 words is due by **Thursday, 11:59 p.m., CST**.
2. For the initial post to be considered substantive, it should be at least 250 words in length and fully cover the topics being presented. Single sentence definitions or responses will not be awarded points.
3. Submit your post by clicking on the **Assignment Link** above, then **Create Thread**. You must create a thread in order to view your peers' posts. Tip: Create your post in a Word document and then copy and paste your work into the thread.
4. A minimum of three (3) responses, **to the original threads of other students**, of 100-200 words each are due by **Sunday, 11:59 p.m., CST**.
5. To view the rubric grading criteria, click on the following link: [Discussion Board Grading Rubric](https://content.bellevue.edu/cst/csd/rubricdbv3.pdf).

**(50 points)**

Hey, Colton! I think you did a fantastic job on your discussion post for this module. You clearly defined telemetry within DevOps and provided what type of data may be collected. You are spot on that telemetry automatically helps teams see what is happening across infrastructure and software. Having data that can help determine where an error started is very beneficial. It saves developers time by directly tracking data and pinpointing where to look for mistakes. If changes are being made or actions are being deployed, there should always be a tracking system that follows it as it runs to document the runtime occurrences.

Hey, Anton! I really enjoyed reading your post for this week's module. Your thorough post covered telemetry, non-Gaussian distribution, and anomaly detection. All three topics have cross points where they help or cover similar issues. As you mentioned, anomaly detection tools help to track non-Gaussian distributions. Telemetry does help a developer team identify errors in a system or software. In my research, I also came across Splunk and New Relic, looking for good anomaly tools. Both Splunk and New Relic seem like good and reliable tools to implement for anomaly detection in DevOps. Teams should consider their size and project demands when choosing the best tool.

Hi, there, Arely! I found your discussion post for this module to be very insightful. You accurately described how Telemetry in DevOps works and provided great examples of what type of data might be collected and why. In a system or other software that can monitor code, resources are saved from being wasted searching for a problem within the whole instead of having a narrowed-down angle view of where to look for the error. You went above and beyond by providing information on non-Gaussian distribution and anomaly detection tools available for use. Based on the information you provided, Dynatrace is the better option over Cisco AppDynamics.