Do you get five nines from your laboratory information system (LIS)? Five nines reliability, or 99.999% availability, is only 5.26 min of downtime over an entire calendar year. Uptime is the gold standard when assessing mission critical, high availability information systems, but it is only one facet of system reliability. What happens when one server fails? Is there reduced functionality or does it cause cascading failure of the entire system? Is the locally present or remotely situated information technology (IT) team ready to address any and all manners of service interruptions? Certainly, the field of Pathology Informatics has long recognized failure is inevitable and thus we add redundancy; redundant power supplies, redundant storage, redundant servers, redundant network connections, etc. The extra expense is significant but is worth the peace of mind the added level of continuity of service provides to our patients. When hardware eventually fails, this redundancy pays off... except, of course, in those remaining, unaddressed sectors where it does not. Compounding this reality is the recurring inconvenient truth that, when a critical piece of hardware fails, it will invariably seem to fail over the weekend and a holiday weekend at that. Murphy’s Law will then kick into ensure that the backup will fail as well. British Airways, one of many examples from the airline industry, dealt with the effects of an IT failure cascade during the United Kingdom’s Spring Bank Holiday in 2017,[1] which in addition to nearly halting operations for several days, resulted in an estimated loss of $129 million in compensation to passengers and incalculable damage to their reputation.

How can this scenario be avoided? When critically examining our contemporary IT stewardship approaches at a systems level, it can be easily seen that we have a blind spot for weaknesses in multiple aspects of our IT infrastructure because we do not adequately test our infrastructure frequently enough, or for some, even at all. In the medical world, the fundamental, and perhaps most ironic, reason we do not test these mission-critical systems is due to patient safety – the production system is sacred, yielding the one and only commandment: “Don’t touch the production environment.” Typically, our backup environments get exercised only occasionally when the production system needs to be taken offline for maintenance. In such circumstances, we (grudgingly) shift our dependency to this secondary environment. Moreover, these scheduled failover activities take place under ideal circumstances, with all critical members of the support team present, prepared and available to assist with any issues that might arise during the downtime activity. In the final evaluation, any seasoned IT steward of laboratory IT systems will have no doubt noted how prevalent it is observed that redundant systems are functionally not equivalent to the primary systems that they support, mostly in an effort to be frugal. For instance, it is more cost-effective to recycle the previous production system into a backup for the next generation production system rather than buying new redundant hardware. We should be profoundly disturbed by this recurring reality, especially given that these systems were replaced for a reason. Moreover, given the frequency with which redundant systems are tested in realistic scenarios, it should not be a surprise that random failures often take place, resulting in outages.

Companies operating true high availability systems do not have an aversion to testing failure scenarios in their production environments. Instead, they embrace them as a source of functional validation of their redundancy strategy and moreover, as a powerful tool for maintaining technical competence of the collective teams expected to triage outages with confidence and rapidity. Netflix, for instance, conducts tests on their customers using an internally developed tool with the given moniker of “Chaos Monkey.”[2] The purpose of Chaos Monkey is to mimic system failures and performance degradation of Netflix’s infrastructure, performed on the production system under typical conditions so that it can identify and reinforce weaknesses. Two key points of that implementation are the use of an opt-out model and the unique timing of testing methodology. Maintainers of specific services can opt-out during periods of significant infrastructure or software change, where it can be reasonably inferred that there is already presumed weakness, owing to an active state of change. Concerning the latter point of timing, since the purpose of Netflix’s testing is to learn as much as possible, testing is only performed on weekdays between 9 am and 3 pm, when people are available to analyze the effects of the induced perturbations.

If we want true, as opposed to theoretical, high availability of the LIS, it may be worth considering testing our primary and redundant infrastructure elements under load in a more meaningful way. Certainly, we are not advocating taking unnecessary and ill-advised risks. Rather, we are suggesting with appropriate site preparation, initial training,
and validation, there is an opportunity to transition the stewardship of our laboratory infrastructure to include the use of an “entropic simian.” This effort may very well pay large dividends in the end, toward the goal of realizing true fault-tolerance and failure resiliency. Until that time, we may be satisfying all the regulatory mavens with our white-glove approach to testing, but truly, we are fooling ourselves if we think that Murphy is buying these gestures as a true preparative measure.

In the authors’ combined 42 years of experience in the Pathology Informatics clinical operational space, including canvassing a fair number of the authors’ contacts at academic- and community-based clinical laboratories, we have not encountered a single instance or anecdote where LIS system managers or support teams have proactively caused an error condition in the live LIS production environment, thereby creating a forced condition where the respective LIS support team absolutely had to respond, real-time, with a definitive and proactive solution. Given this methodology has already become the standard of practice in other, nonmedical IT sectors, there is merit in exploring its appropriateness in the pathology/laboratory medicine space. At present, there are numerous pathways by which a clinical laboratory can attain accredited status, including oversight by the College of American Pathologists and the Joint Commission. It is interesting to note that at present, no regulatory oversight body requires, or even mentions, engineered failure modes as a component of meeting certification requirements. This absence, in and of itself, is prima facie evidence that this approach has not yet surfaced as a technique on merit in the laboratory medicine space.

Financial support and sponsorship
Nil.

Conflicts of interest
There are no conflicts of interest.

Christopher L. Williams¹, David S. McClintock², Ulysses G. J. Balis²

¹Department of Pathology, University of Oklahoma, Norman, OK,
²Department of Pathology, Division of Informatics, University of Michigan, Ann Arbor, MI, USA

Address for correspondence: Dr. Ulysses G. J. Balis, 1301 Catherine Street, 4233A MSI, Ann Arbor, MI, USA.
E-mail: ulysses@med.umich.edu

References
1. A Computer Failure at British Airways Causes Chaos. The Economist Website. Available from: https://www.economist.com/blogs/gulliver/2017/05/going-nowhere. [Last accessed on 2018 Jan 29].
2. DevOps Case Study: Netflix and the Chaos Monkey. DevOps Blog. Available from: https://www.insights.sei.cmu.edu/devops/2015/04/devops-case-study-netflix-and-the-chaos-monkey.html. [Last accessed on 2018 Jan 29].

This is an open access article distributed under the terms of the Creative Commons Attribution-NonCommercial-ShareAlike 3.0 License, which allows others to remix, tweak, and build upon the work non-commercially, as long as the author is credited and the new creations are licensed under the identical terms.

How to cite this article: Williams CL, McClintock DS, Balis UG. The case for an entropic simian in your laboratory: The case for laboratory information system failure scenario testing in the live production environment. J Pathol Inform 2018;9:7.
Available FREE in open access from: http://www.jpathinformatics.org/text.asp?2018/9/1/7/228967