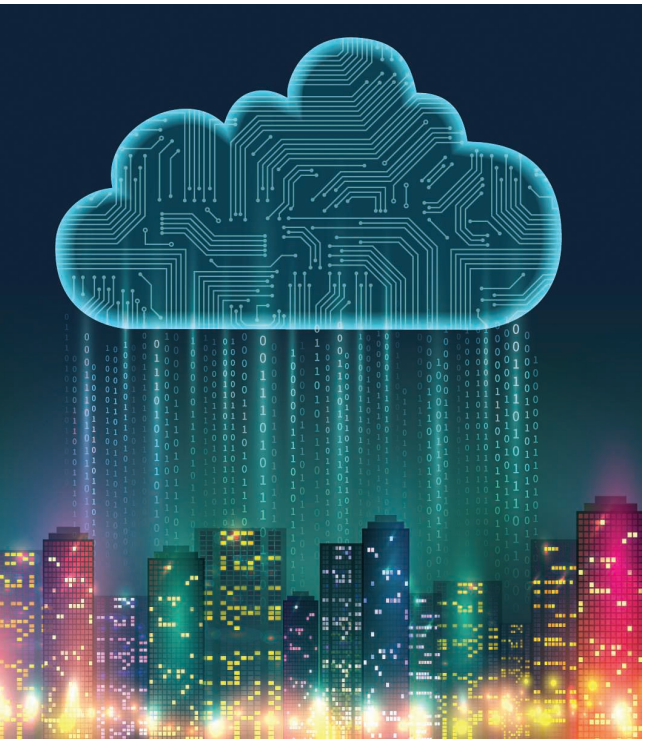
The Soft Underbelly of Cloud Security

Ron Herardian | Basil Security

Cybersecurity, regulatory compliance, and cloud operations are interdependent. The operational security of the cloud-operations function is the weakest link in cloud security. This problem is not well recognized, and new technologies are needed in the face of current security and regulatory trends.



People tend to view themselves as having control or influence over situations that, in fact, are chance events.[1] The illusion of control explains why the operational security of the cloud is often a blind spot, despite our best efforts to establish proper security policies and controls. Unless there is technical enforcement and provable accountability, establishing security policies and performing compliance-related audits serve only to foster the illusion of control.

Digital Object Identifier 10.1109/MSEC.2019.2904112
Date of publication: 14 May 2019

Operational Security Is Failing

Cloud-security solutions and tools exist for different infrastructure layers and for different steps in continuous integration/continuous delivery (CI/CD) pipelines as well as in different cloud-provider environments. Organizations manually stitch disparate security tools and services together and then attempt to implement security policies based on their particular infrastructure and application stack. In general, there is no single security policy framework that operates at multiple infrastructure layers and across private cloud, hybrid cloud, and multicloud environments.

No single security policy framework governs the range of tools used by development and operations (DevOps). DevOps tools include, but are not limited to, infrastructure as code, configuration management (for example, Chef and Puppet), and orchestration systems (for example, Kubernetes). The unification and extension of security policy enforcement from on-premise systems to the cloud, and across multiple cloud providers, is a largely unsolved problem. In particular, the operational security of the cloud-operations function is the weakest link in cloud security and the largest single risk area as illustrated by the following:

- 72% of organizations have no visibility into IT staff activity.[2]
- 28% of all cyberattacks are insider attacks.[3]
- 44% of data breaches are attributable to insiders.[4]
- 90% of internal bad actors displayed no worrying characteristics prior to their attacks.[4]
- 58% of patient health information data breaches are caused by insiders.[5]
- Attacks by malicious criminal insiders are costlier than system glitches and negligence.[6]

The risk and potential damage of insider attacks are often disproportionate to the typical responsibility and experience level of DevOps personnel.

Insufficient Controls

While security models, policies, operational procedures, and change-control processes define what should and should not be done, there is a general absence of technical security policy enforcement and provable accountability. DevOps personnel, for example, can typically run any tool or command in any environment; for example, they may use the Kubernetes `kubectl` command to push `configmap` updates. Although `configmaps` may be under source control (for example, using `git`) and

 Copublished by the IEEE Computer and Reliability Societies 1540-7993/19©2019IEEE

subject to pull-request approvals, operators can often modify files and immediately deploy them, bypassing source control. Similarly, DevOps personnel with access to multiple environments can easily make changes in the wrong environment.

When DevOps personnel use the `kubectl` command to manage Kubernetes clusters, they must have access to the certificates used to secure the Kubernetes application programming interface (API) endpoints. These certificates (and similar secrets) could easily be sold on the dark web in exchange for cryptocurrency, quickly leading to large-scale exploits, such as infrastructure-as-a-service cryptojacking. Attribution is often problematic when security incidents occur. Commands like `kubectl` and other DevOps tools may be run locally or on remote hosts, but vulnerabilities remain. In practice, the risks are largely ignored; for example, emphasis is placed on restricting access to secrets and following proper procedures with the threat of disciplinary action. While efforts to integrate development, security, and operations (DevSecOps) are leading to new security models, practices, and tools, the underlying problems of technical security policy enforcement and provable accountability remain.

DevSecOps: Dead on Arrival?

CyberArk[7] describes some reasons for the previously cited shocking statistics, noting that most businesses do not understand that privileged accounts and secrets exist in multiple systems and cannot identify where all privileged accounts and secrets are. Use of the public cloud, software as a service, and DevOps are growing quickly, and security programs cannot keep pace. DevOps and security teams do not consistently work together and are not well integrated throughout the development pipeline.

DevSecOps is the putative answer. Many security product and service vendors claim their offerings facilitate the transition to DevSecOps. However, distributing security responsibility and fostering a culture of security and accountability throughout the development pipeline are ultimately a set of best practices. DevSecOps proponents generally reiterate well-known best practices (for example, leveraging identity and access management, implementation of "least privilege," segregation of duties, security code reviews, and so forth). Automated vulnerability and malware security scanning throughout the CI/CD pipeline is an established best practice.

Security orchestration automation and response (SOAR) is a significant improvement that works well with a DevSecOps approach. However, neither SOAR nor DevSecOps adequately addresses the operational security of the cloud operations function. Responses to security related information and events occur after the fact. A better approach would evaluate the current state of security policies and running systems whenever a change is attempted. The problem is that there are no viable technical solutions to do so. Doubling down on insufficient methodologies without substantive technical backing encourages the illusion of control.

Quis Custodiet Ipsos Custodes?[8]

Consider System and Organization Controls Type 2 (SOC 2) as an example; the controls governing security, availability, processing integrity, confidentiality, and privacy are a combination of prescriptive controls, descriptive controls, and test results. Prescriptive controls reference security architecture and policy documents, operational procedures, and change-management protocols, along with records of associated authorizations and actions.

In most organizations, prescriptive controls are, essentially, documents. Descriptive controls involve technical systems and tools that provide security administration, management, monitoring, reporting, and alerting. The SOC 2 audit process examines the completeness and adequacy of prescriptive controls while the associated evidence of compliance is provided by descriptive controls.

In general, prescriptive controls lie within the purview of management while descriptive controls lie within the purview of the cloud-operations function. In preparation for the SOC 2 audit process, for example, the cloud operations staff generate documents and institute technical systems to meet audit requirements. In the worst case, evidence of compliance can be manufactured or even falsified so as to feign compliance during audits while ignoring controls at other times.

Figure 1 illustrates the different views of the relationship between security and compliance.

1. *Management view*: This view consists largely of prescriptive policies and procedures, compliance with which is a condition of employment.
2. *Auditor view*: This view encompasses the management view, documented evidence of compliance, and (potentially) test results.
3. *DevOps view*: From this view, cloud operations staff control live systems applications and data at more than one level of the infrastructure and application stack. Related job functions typically involve access to product and infrastructure code, secrets, and environments. While policies and procedures dictate what is and is not allowed, typically there are few, if any, mechanisms to prevent unauthorized actions before the fact.

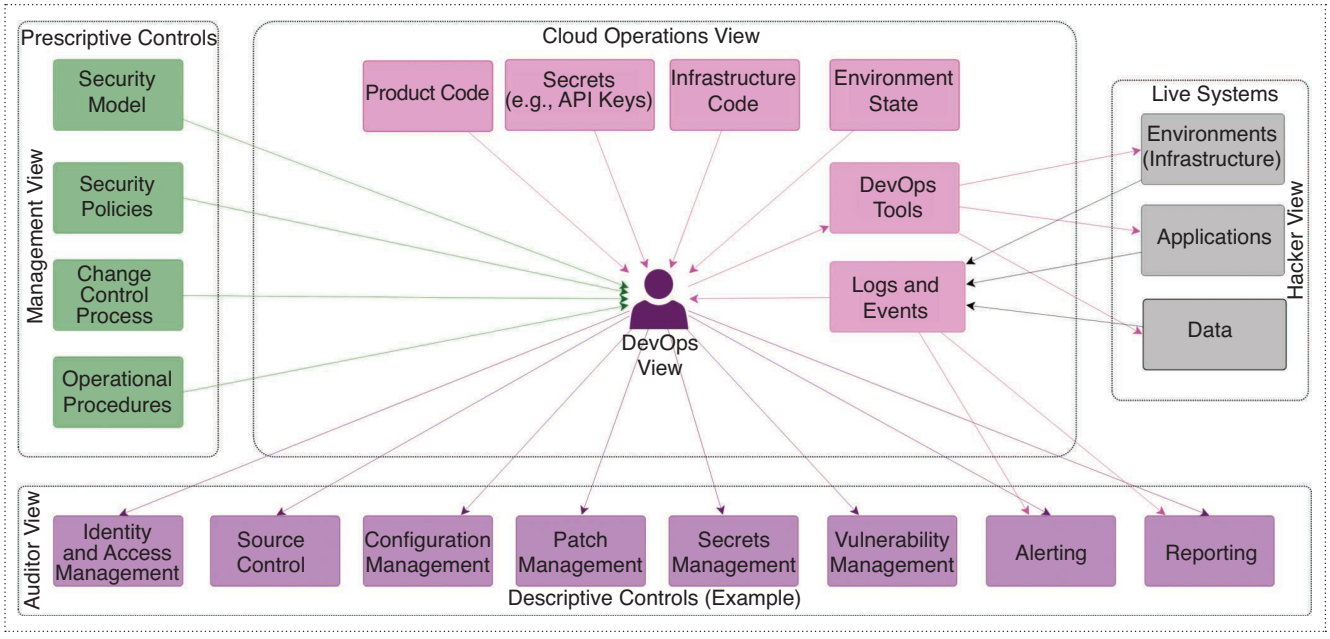


Figure 1. Disparate views of policy and enforcement.

4. *Hacker view*: A hacker view of a system is unrelated to the management and auditor views. Security policies and operational procedures are essentially irrelevant. The hacker seeks access to applications, secrets, and data. A hacker is principally concerned with discovering any weakness missed by the security and operations teams.

The Honor System, the General Data Protection Regulation, and the California Consumer Privacy Act

In most organizations, operations workloads and staffing levels necessitate that operations personnel be as interchangeable as possible. DevOps personnel must be able to access environments and secrets and to read, modify, and run code (for example, infrastructure as code, custom programs using cloud provider APIs, and configuration management code) to make required changes. In the worst case, operations personnel can make changes in any environment; directly access secrets (credentials, API keys, certificates, encryption keys, and so forth); and access, modify, and run code without review.

It is commonly assumed that operations personnel follow policies and procedures, but there is generally no technical capability to enforce documented controls and attribution, at least without unified audit logging. An organization's recourse upon discovering errors or omissions related to improper procedures, code changes, and malicious activity is to terminate the responsible individuals, but, at that point, the damage is done.

Operational security of the cloud-operations function is coming into focus in Europe,[9, 10] as evidenced by General Data Protection Regulation (GDPR) fines signaling out cases where policies were not enforced and where a lack of accountability was endemic. Similarly, the passage of the California Consumer Privacy Act (CCPA) in the United States indicates that a trend is forming. Organizations that fall victim to insider threats may also be fined by regulators. Albert Einstein famously said that the definition of insanity is "doing the same thing over and over and expecting different results" and that "we cannot solve our problems with the same level of thinking that created them." An operational model based on the honor system provides neither security nor compliance. A belief that it does suffers from the illusion of control. New technologies are needed to properly meet requirements stemming from regulations such as the GDPR and the CCPA.

Are Zero-Trust Operations Possible?

The zero-trust model of information security was first proposed by John Kindervag, senior analyst at Forrester Research, Cambridge, Massachusetts, in 2010.[11] Kindervag argued that the biggest issue facing security and risk professionals was that the traditional trust model had broken down and that threats from malicious insiders were growing at a rapid pace. In the following years, zero trust was implemented for network security, and it has been shown to prevent data breaches.[12] Zero trust has not been applied to the operational security of the cloud-operations function because the necessary

technology has not been developed. In particular, the following innovations are needed:

- the ability to define security policies in software
- technical security policy enforcement at runtime
- audit logging for all actors and actions
- better protection of secrets and environment information.

Software-defined security is needed to dynamically control who can run which tools and code in which environments using which secrets and under exactly what conditions. Technical policy enforcement means that policies are enforced in real time, preventing unauthorized actions rather than merely detecting them after the fact. Policies must apply at different levels of the infrastructure and application stacks and in different environments. Audit logging, such as for change control, must provide accountability and allow attribution with 100% accuracy. Secrets, such as credentials, API keys, encryption keys, and certificates, should not be accessed directly by DevOps personnel. A secrets-management solution should be put in place, and strong encryption should be used to prevent environment-related information and secrets, including any data in log files, from being exposed to DevOps personnel.

Data related to insider errors and malicious insiders reveal that the operational security of the cloud-operations function is the weakest link and the highest-risk area in cloud security. New regulations require enterprises and service providers to conduct operations in a more secure manner and maintain records that establish provable accountability. Current best practices, including DevSecOps, are insufficient to address the problem. New technologies are needed to define security policies in software; provide technical security policy enforcement proactively; provide audit logging, including all actors and all actions; and better protect secrets and sensitive information about cloud environments and applications. ■

References

1. E. J. Langer, "The illusion of control," *J. Personality Social Psychol.*, vol. 32, no. 2, pp. 311–328, 1975. [Online]. Available: http://dx.doi.org/10.1037/0022-3514.32.2.311
2. J. Melnick, "Cloud security risks and concerns in 2018," Netwrix Blog, Jan. 23, 2018. [Online]. Available: https://blog.netwrix.com/2018/01/23/cloud-security-risks-and-concerns-in-2018/
3. Verizon, "2018 data breach investigations report public sector excerpt," 2018. [Online]. Available: https://enterprise.verizon.com/resources/reports/2018/2018_dbir_public_sector.pdf
4. PwC, "PwC audit committee update: Insider threat," 2018. [Online]. Available: https://www.pwc.co.uk/audit-assurance/assets/pdf/insider-threat-for-google.pdf
5. Verizon, "2018 protected health information data breach report," 2018. [Online]. Available: http://www.verizonenterprise.com/resources/protected_health_information_data_breach_report_en_xg.pdf
6. Ponemon Institute and IBM Security, "2017 cost of data breach study," 2017. [Online]. Available: https://www-01.ibm.com/common/ssi/cgi-bin/ssialias?htmlfid=SEL03130WWEN
7. CyberArk, "Unaware and unprepared: DevOps secrets at risk," 2018. [Online]. Available: https://www.cyberark.com/resource/unaware-unprepared-devops-secrets-risk/
8. Juvenal, Satire VI, lines 347–348. "Sed quis custodiet ipsos custodes?" ("But who will guard the guards themselves?")
9. O. Schmidt, "Germany's first fine under the GDPR offers enforcement insights," International Association of Privacy Professionals, Nov. 27, 2018. [Online]. Available: https://iapp.org/news/a/germanys-first-fine-under-the-gdpr-offers-enforcement-insights/
10. A. Menezes Monteiro, "First GDPR fine in Portugal issued against hospital for three violations," International Association of Privacy Professionals, Jan. 3, 2019. [Online]. Available: https://iapp.org/news/a/first-gdpr-fine-in-portugal-issued-against-hospital-for-three-violations/
11. J. Kindervag, "No more chewy centers: The zero-trust model of information security," Forrester Security Forum, Boston, MA, 2010. [Online]. Available: http://crystaltechnologies.com/wp-content/uploads/2017/12/forrester-zero-trust-model-information-security.pdf
12. CloudFlare, "Zero-trust security: What's a zero-trust network?" 2019. [Online]. Available: https://www.cloudflare.com/learning/security/glossary/what-is-zero-trust/

Ron Herardian is the cofounder and chief executive officer of Basil Security, Atlanta, Georgia. His current research interests include practical applications of blockchain technology, security policy as code, and stateful security policy enforcement. Herardian received an M.L.A. with a concentration in intellectual history, including history and philosophy of science, from Stanford University, California. He is a Member of the IEEE and a senior member of the Association of Computing Machinery. Contact him at ron@basilsecurity.com.

IEEE COMPUTER SOCIETY DIGITAL LIBRARY
Access all your IEEE Computer Society subscriptions at **computer.org/mysubscriptions**