WHITEPAPER

Automation for Security and Security for Automation: Finding Symbiosis in DevOps

NOPSEC

nopsec.com | hello@nopsec.com

Table of Contents

Executive Summary

IT automation tools are proving to be the foundation for a profound shift in how IT services and applications are built, delivered and maintained. Tools such as Chef, Puppet, Ansible, and Jenkins have helped to fuel the DevOps trend and enable IT Operation teams to lower the time and resources consumed by manual procedures. Now, security practitioners are evaluating the potential to replicate the automation model embraced by their operations and developer counterparts to streamline existing vulnerability management and remediation processes – which are frequently fraught with inefficiencies.

Automation tools, however, are not a panacea if the vulnerability management process itself remains plagued by information overload and little or no internal context of the risk posed to valuable assets by vulnerabilities. Automation can be a boon, if organizations can get smarter about prioritizing remediation efforts via analytics and risk-driven prioritization, maintaining seamless integration between automation tools and tying vulnerability management workflows to a current mapping of assets.

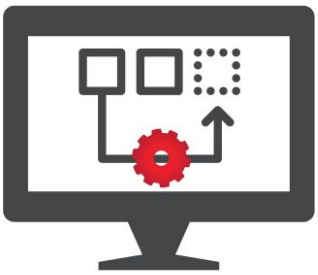
With a common set of tools in place – especially for configuration management and orchestration – and integration with ticketing and workflow management, information security professionals can foster deeper collaboration with IT operations and drive consistency. And, security professionals can realize the potential of embedding vulnerability remediation and patching

processes into the software development lifecycle through integration with release management and continuous integration tools.

Introduction

Despite the growing skepticism with which the phrase 'DevOps' is now greeted, even the most conservative IT departments are finding value in IT automation tools that may fall under the DevOps marketing rubric. Certainly, the use of a common set of automation tools can bring together multiple functions within an enterprise to improve operational efficiency and reduce time and resources spent on meeting compliance requirements.

But in order to seamlessly implement automation, organizations need to find ways of managing the sheer volume of vulnerability management tasks, reduce information overload from vulnerability scans and establish a more risk-centric approach to pinpoint the highest priority vulnerabilities in the context of an organization's infrastructure and assets. A critical vulnerability on a non-production host that does not process sensitive information does not represent as much risk as the same vulnerability in an Internet-facing service, for example.



IT automation has two facets: leveraging tools for the automation of manual procedures and integrating frictionless remediation for maintenance, configuration management or software delivery.

No doubt, reducing the time and effort spent on meeting compliance requirements can be of value to many organizations. Since meeting compliance requirements is such a resource-intensive and inefficient process for many

organizations, many struggle with how to prioritize their activities to better address risk. Less time spent on activities with a low-value return can translate into more scope for greater focus on improving security posture and addressing vulnerability risk through frictionless remediation.

But IT automation alone cannot resolve the information overload challenge that bedevils many vulnerability management programs as a result of the overwhelming output from vulnerability scans. Adoption of automation tools doesn't necessarily result in greater visibility into the state of applied patches for instances and transparency into change management processes. What this means is that adopting IT automation tools (whether as part of a DevOps cultural shift or independently) is a partial answer to driving more efficient workflows and processes.

For vulnerability management, IT automation has two facets: leveraging tools for the automation of manual procedures (such as applying and installing patches and configuration settings) and integrating frictionless remediation into an IT automation workflow for maintenance, configuration management or software delivery. In both scenarios, deriving value from automation is contingent on the effectiveness of vulnerability and remediation workflows focused on the riskiest and most urgent vulnerabilities, coupled with seamless patch installation.

IT Automation - Not Just for the Cool Kids

The Chicken or the Egg? DevOps and IT Automation

The term DevOps has become so amorphous and widely applied that it's challenging to come up with a definitive description. However, the term can be understood as a set of practices that are cross-functional in nature, and span multiple roles within an organization that have traditionally operated within discrete silos. DevOps has as much to do with a cultural shift away from the mentality of 'throw it over the wall' as it does with a specific set of technologies.

In practical terms, then, it's useful to draw a clear distinction between DevOps practices and the set of tools that enable DevOps through automation of manual steps. DevOps tools tend to get lumped together, but they can serve multiple purposes, address different constituencies and automate different processes.

DevOps tools can be effective for orchestration of deployment activities that span functional roles and environments, or enable automation of infrastructure provisioning in response to business needs. In other instances, DevOps tools can be used to define 'playbooks' that package and distribute configuration management settings for multiple hosts, removing the need for custom scripts and manual intervention. In the area of application delivery, source control repositories such as Jenkins facilitate Continuous Integration (CI) and Continuous Delivery

Assigning all critical vulnerabilities with equal risk scores has the practical outcome of prioritizing none.

pipelines to compress the time between release cycles, integrate testing cycles, and automate new software releases.

Although not specifically intended for the purpose, the IT automation tools that underpin DevOps practices can be applied to the discipline of vulnerability management. For security practitioners, automation tools can reduce the number of repetitive manual steps using the same principles as IT operations put into practice for configuration management, for example. And, because remediation efforts can be implemented through a structured change management process, automation tools can improve the consistency.

IT Automation for Vulnerability Remediation and Management

Reducing the attack surface is a cornerstone principle of information security. By definition, remediating software vulnerabilities will help reduce the attack surface – which goes to explain why so many compliance frameworks incorporate requirements for a vulnerability management program. Also, there is a strong correlation between the time taken to remediate a vulnerability and the relative risk it poses as an exploit vector. According to a research conducted by Malwarebytes, zero-day vulnerabilities are weaponized faster than ever before, with cyber criminals now developing zero-day exploits within 4 days -- half the time than previously observed.

There is, however, a significant distinction between meeting compliance requirements and taking a risk-centric approach to

reducing the attack surface. Certainly, critical vulnerabilities can be an urgent priority for remediation, since they can represent low-hanging fruit for opportunistic hackers using automated tools to execute exploits. The rise of the cybercrime economy means that tools are widely available to spot vulnerabilities and exploit them without requiring an enormous amount of technical sophistication on the part of an individual attacker.

But even for critical vulnerabilities, the likelihood of an attack as well as an exploit based on the vulnerability should be evaluated in the context of an organization's assets. If executing the exploit doesn't yield anything of material value to the attacker, the relatively low level of effort required means they'll quickly move on to the next target. Assigning all critical vulnerabilities with equal risk scores has the practical outcome of prioritizing none. Without a set of priorities driven by insight into the relative risk of vulnerabilities, vulnerability management efforts are effectively hamstrung – and IT automation efforts are likely to fail or fall short.

Equally, with an effective vulnerability management program in place that emphasizes risk based on the vulnerability in tandem with frictionless remediation, automation can deliver significant value in both operational efficiencies and risk containment. With formal and programmatic integration with service management and ticket management systems, for instance, operations and security teams have greater visibility into the state of systems where patches have not been applied – amounting to an un-remediated vulnerability.

The benefits of integrating vulnerability management with IT automation workflows are best realized when focused on efficiently reducing the attack surface. This is accomplished by identifying and prioritizing the most urgent vulnerabilities, determining the assets with the riskiest exposure through analytics, and integrating this intelligence into a closed loop remediation process. Automation is an integral component of driving operational efficiency, but should be viewed as a piece of the puzzle for focusing remediation efforts on the riskiest vulnerabilities based on an organization's assets rather than a structural fix. Integration of threat intelligence provides an additional layer of context to optimize vulnerability management workflows.

Moving to a Collaborative Model - It Takes a Village to Securely Automate

Building Bridges: Improving Remediation Precision

Automation is an integral component of driving operational efficiency, but should be viewed as piece of the puzzle for focusing remediation efforts on the riskiest vulnerabilities.

According to the SANS Institute, "the main objective of a vulnerability management process is to detect and remediate vulnerabilities in a timely fashion."

In other words, the driving impetus for a more efficient vulnerability management program is that time is not on your side. In order to best marshall finite resources in the race against time, organizations need a better and more systematic approach to framing why some vulnerabilities should be remediated over others, and defining how they should be remediated in a way that is frictionless both for security professionals and their operations

counterparts – especially as automations tools become more broadly adopted. From the perspective of reducing risk and driving more efficient workflows, vulnerability risk management and IT automation are two sides of the same coin.

To reiterate, deriving the maximum benefit from automation efforts and improving collaboration requires breaking from a traditional remediation strategy. A shift must take place from a strategy based on lowering vulnerability counts to one driven by risk.

In a similar vein, breaking the bottleneck of quarterly vulnerability scans and moving to a continuous vulnerability detection model requires both more precision and alignment with asset management and discovery. After all, the challenge is not only determining what changes and patches need to be applied, but also to what systems, hosts or other targets. To foster collaboration, security teams need to set priorities in order to lessen the ongoing conflict over resources that their operations face as well.

Progressing Toward the Easy Button

Automation can prove to be an effective tool to bridge the divide between operations who see vulnerability management as a discrete activity and security that view it as a series of related tasks. However, as it is the case with the use of IT automation tools in the context of what could be called DevOps processes, there needs to be intelligent integration of vulnerability

management workflows with automation tools and existing infrastructure management.

For example, for many organizations, Microsoft SCCM is the engine for configuration management – even tools such as Chef, Ansible, and SaltStack rise to prominence. In order to facilitate the transition between remediation by security professionals and configuration management by operations, it's important for security professionals to be able to stage changes and necessary patches in configuration management tools so that they can be applied once operations have approved the changes.

Likewise, security professionals can compile a 'playbook' using tools such as Ansible that can automate one change applied to many systems. These changes can be staged with a set of pre-loaded available patches and recommended configuration settings which operations can evaluate before running. Staging can also be done through a set of Powershell scripts applied by the SCCM packages, which can be reviewed before running.

Automation is not without its risks, however. Systems availability in production can be adversely affected by bad patches, failed pushes, bugs, or even fragile or unpatchable systems. In this sense, ensuring that current inventory of assets is in place – and that vulnerability management systems can apply values to assets as they are onboarded either through an API or some other mechanism – is crucial to collaboration. As organizations adopt public cloud services more broadly, this requirement will only become more crucial.

Accountability is Collaboration's Good Friend

Automated workflow, integrated ticketing systems and incident management allow IT teams to stay connected throughout the remediation lifecycle.

Improved collaboration is not only the outcome of better prioritization of vulnerability remediation efforts and use of common tools. Organizations also require improved visibility and a consolidated view into the state of vulnerability risk management. Integration with service management – whether in the form of ServiceNow, BMC Remedy, or Atlassian Jira – should be seen as a means to an end of improved visibility into the state of the remediation process and the foundation for collaboration.

Determining the state of change management is integral to evaluating risk and exposure. Extracting information specific to remediation change requests and closed tickets and then reporting on state in a consolidated dashboard is not only important for compliance purposes – but also to determine whether patches have been appropriately installed. Automated workflow, integrated ticketing systems, and incident management in conjunction with integration into asset management and configuration management tools allow IT teams to stay connected throughout the remediation lifecycle and deliver up-to-date and comprehensive reporting on current status.

A single, consolidated view can also help to improve communication and refine workflows – so as to avoid the 'tossing over the wall' syndrome that can plague remediation processes. In any organization, there is a tendency to focus on the easy tasks first – which may not have any direct relationship

on reducing risk. Equally, security professionals may need to provide additional insight into a remediation or patch.

Faster, and Lower Risk - Integration of Vulnerability Management Ops and Devops

Faster Deployment Cycles - and Improved Security

Continuous Integration (CI) and Continuous Delivery (CD) models have significant implications for how vulnerability management is delivered and remediation workflows are integrated into the software delivery pipeline. Ideally, implementing CI/CD means that build, test, and deploy processes that would have taken months to reach a conclusion in the past are reduced to weeks or even days – with greater consistency. There is the argument that the ability to apply incremental changes means that remediation is less of a massive task, since not all the changes to the code need to be implemented all at once. And, since changes can be implemented more quickly, vulnerabilities can be patched far more quickly, and earlier in the development process. Again, CI/CD can answer the question of how remediations can be better automated – but does not address the question of why and how much.

In fact, as organizations look to embrace DevOps principles, the challenge of information overload in managing and remediating vulnerabilities is compounded. In order to effectively implement agile development processes in tandem with automation of

Security professionals must deliver actionable and specific information that can be easily incorporated into the automation workflow.

vulnerability remediation, the onus is on security to better pinpoint actionable vulnerabilities in code and modules as they move the software delivery pipeline.

Equally, speed comes with its risks: the use of IT automation tools can also help proliferate software vulnerabilities at higher velocity, and move insecure code into the production environment far quicker than before. The challenge faced by security practitioners is enabling developers to leverage these tools, and identify strategies to improve collaboration to ensure that use of DevOps automation tools does not come at the cost of new security risks.

Vulnerability Management as DevOps Practice

Certainly, programmatic integration via RESTful APIs with CI/CD tools (such as Chef or Puppet) is important to facilitating cross-functional collaboration. However, in order to ensure a cultural fit for security and align remediation with the impetus behind adoption of CI/CD, vulnerability management output must be framed in terms of risk. This involves not only improved filtering of output scans to deliver a set of remediation priorities but also the incorporation of threat intelligence. Since CI/CD processes incorporate third party libraries and open source components, traditional vulnerability scans, and even penetration tests can overlook the threat posed by unpatched open source components.

Ideally, developers, testing, and operations teams treat vulnerability information and remediation workflows as they would any other application development issue. While an un-remediated vulnerability doesn't necessarily bear the equivalent risk as a potential performance or functionality bug from the perspective of those teams, remediation should be seen as a necessary step and integral to the delivery workflow.

In turn, that means security professionals must deliver actionable and specific information that can be easily incorporated into the automation workflow. Providing a long list of vulnerabilities is both inefficient and counterproductive – and can short-circuit automation processes. By integrating functional, integration, and information security testing as a combined process into the daily operations, defects and vulnerabilities can be identified and remediated more responsively.

Conclusion

While IT automation tools can contribute toward making vulnerability and remediation processes more efficient, consistent, and responsive than before, they can also introduce risk in other contexts. In the absence of adequate controls for moving applications to production and greater cooperation between information security and their IT operations and development counterparts, there is a risk that vulnerabilities move into production code and systems at an accelerated rate. Inefficient vulnerability management processes introduced into a

set of processes designed for speed places security and other constituencies on a collision course.

Still, the benefits are clear of melding IT automation with vulnerability risk management: rather than wait for the results of quarterly vulnerability scans mandated by compliance requirements, organizations can move to continuous vulnerability detection and remediation with automation in place. In conjunction with frictionless remediation, integration with automation tools and configuration management platforms such as Microsoft SCCM can reduce the overhead spent on meeting compliance requirements and improving risk posture.

Automation in and of itself cannot fix broken or inefficient vulnerability management processes. Certainly, time-consuming, low-value, and repetitive vulnerability management tasks can be automated, and remediation processes can be integrated into software delivery workflows centered on source control repositories. However, without a clear understanding of priorities based on risk, automation can still amount to an iteration of existing processes that are bedeviled by too much information and not enough context to make it actionable.

Find out how NopSec's Unified VRM can help you think like a hacker and stay ahead of the trends. Visit www.nopsec.com or email hello@nopsec.com for additional information or to request a demo.

About NopSec

NopSec operates with one mission: to help people make better decisions to reduce security risks. Our team is passionate about building technology to help customers simplify their work, manage security vulnerability risks effectively, and empower them to make more informed decisions. Our software-as-a-service approach to vulnerability risk management offers an intelligent solution to dramatically reduce the turnaround time between identification of critical vulnerabilities and remediation.

NopSec helps security professionals simplify their work, effectively manage and prioritize vulnerabilities, and make better informed decisions.

NopSec's Unified VRM is an innovative threat and vulnerability management solution that addresses the need for better prioritization and remediation of security vulnerabilities in a single platform.

NopSec Inc. • www.nopsec.com • info@nopsec.com