VENAFI®

SECURING DEVOPS AT THE SPEED OF BUSINESS

www.venafi.com

All DevOps practitioners need to understand that they share important responsibilities for security and that it isn't just someone else's job.

Introduction

Continuous improvement and innovation are two fundamental principles any business needs to stay alive in today's digital world. The pace at which an organization must deliver on these two principles is accelerating greatly. Software and information technology (IT) services play an enormous role in enabling the modern enterprise to create new products, service customers, refine new processes, and develop new business models more effectively and efficiently than ever before. This agility is largely the result of organizations integrating product development with their IT operations, in what has become known as DevOps or Fast IT.

This paper discusses how automating DevOps security can accelerate the pace of innovation. When speed and security coexist in a DevOps environment, organizations can bring products and services to market faster than ever before. We also show the benefits of standardizing and automating the insertion of keys and certificates throughout the process. Plus, we include examples of how automation works with some popular DevOps tools and platforms for built-in security.

DevOps Has Arrived and Is Making Its Mark

By adopting a DevOps philosophy, organizations closely align software development with IT operations to optimize product and service delivery throughout the organization. In practice, this is about reducing time to market while maintaining quality and reliability. It is no surprise, then, that DevOps is quickly becoming the de facto model for continuous improvement and innovation for today's leading organizations. Accordingly, a global study by CA Technologies found that companies that embraced a DevOps methodology increased their speed to market by 20%, leading to a 22% boost in customer relations and a 19% increase in revenue.[1]

By taking advantage of agile software development methods, IT virtualization technologies, and new DevOps platforms, such as Chef, Puppet, Docker, and HashiCorp, organizations are able to automate much of the process of creating, testing, and releasing new offerings quickly and inexpensively. But they may do so at a cost: speed often increases the risk of compromised security.

DevOps Focuses on Speed, Not Security

The challenge many organizations have discovered, however, is that not every process in this new world of DevOps is fast and automated. Take security as an example. The procurement and provisioning of encryption keys and digital certificates required to ensure secure communication across the DevOps environment remains largely a slow and manual endeavor that runs counter to the fast DevOps philosophy. Certificates are essential for protecting pre-production code from attacks, especially as it moves across the environment. Certificates also ensure that code remains protected through its final delivery. Yet, many DevOps teams simply ignore or sidestep certificates throughout the process.

[1] CA Technologies. *TechInsights Report: What Smart Businesses Know About DevOps.* 2013.

When much of the DevOps process is automated, disrupting the workflow with manual certificate processes is often seen as something developers would rather avoid.

But the speed of DevOps does not have to come at the expense of security. Organizations are finding they can centralize, standardize, and automate the process of procuring keys and certificates as part of the end-to-end DevOps environment. Automation enables DevOps teams to deliver applications and IT services more effectively and more securely. But to do this, DevOps practitioners need to understand that they share important responsibilities for security; it isn't just someone else's job.

Fast IT and Slow IT

Established companies have an array of legacy applications they maintain to run their business, and not all are well suited to a fast-paced DevOps environment. Many critical tier-one applications, for instance, must support a minimum of five-nines (99.999%) availability, and the notion of continuous upgrades is still too risky for these types of applications. To address this, businesses are creating a bifurcated environment where traditional (slow) IT will coexist with the fast and nimble DevOps IT. Gartner recommends CIOs heavily promote the mentality of a digital startup within their traditional organizations, where one group supports existing applications that require stability and another (DevOps) delivers "Fast IT" for more innovative projects.[2]

While many CIOs are pursuing such bi-modal IT strategies, current security practices for both are still closely aligned with the traditional IT model. Hence, security is often considered an afterthought, when it should be designed into the process from the beginning. This is where the intersection of security and DevOps breaks down because the slow and manual method of "bolting on" security to applications at the end of the project does not align well with DevOps, which assumes security is built into the process.

Traditional Security is Inadequate

Adopting traditional security for this new DevOps model has become somewhat problematic; in part, because manually procuring and provisioning keys and certificates is simply too slow. Because the DevOps process is automated, disrupting the workflow with manual certificate processes is something developers would rather avoid. In fact, many developers find it easier to ignore certificates to keep security from delaying the release.

Dismissing or implementing keys and certificates poorly can not only expose the organization to unnecessary security vulnerabilities, but it can also lead to chaos by inserting inconsistent, manual steps into an increasingly automated environment. Rather, the goal is to provide full control and automated, standardized security over the source code repositories and the DevOps frameworks developers use every day, so no changes can be made by unauthorized persons.

[2] Gartner *The Four Steps to Manage Risk and Security in Bimodal IT.* September 2015. Doc: G00297713

When asked if the speed of DevOps makes it more difficult to know what is trusted or not in their organizations, 79% of CIOs admitted that it does.

The Dangers of DevOps Bypassing Security

Not maintaining control over the environment can enable attackers to use a compromised, stolen, or forged certificate to impersonate, eavesdrop, and monitor a target's infrastructure, cloud, or mobile devices and decrypt communications thought to be private. Misuse of keys and certificates can result in costly:

- **Data breaches**
- **Application outages**
- **Failed audits**
- **Lack of compliance**

When asked if the speed of DevOps makes it more difficult to know what is trusted or not in their organizations, 79% of CIOs admitted that it does.[3] As the speed of IT increases with the elastic creation and decommissioning of services, keys and certificates will grow in orders of magnitude. Managing this vast number of keys and certificates at the speed and scale of DevOps will become even more critical.

The use of development containers is increasing. Within the next two years, 67% of businesses plan to use them in production environments. Even with this increase in adoption, 60% said they still have security concerns.[4] Ultimately, developers know they need to maintain a secure environment and development process. So why is it that many simply are not doing it?

Building Security into the DevOps Process

DevOps practitioners have been trained and rewarded for delivering software updates quickly and continuously and often bring their favorite coding tools with them. To be effective, security has to work well with these tools.

However, developers are typically not security experts and most do not aim to be. They are less likely to consider anything but the most basic security measures to meet their objectives, especially given past struggles with the error-prone manual nature of certificate provisioning.

The result is that security is either ignored completely or minimized to maintain the velocity of DevOps. The following list represents some of the ways DevOps teams get around the proper deployment of certificates:

- Don't use TLS/SSL to secure connections
- Create their own Certificate Authorities
- Create self-signed certificates
- Create certificates with weak signature algorithms
- Misinterpret or completely ignore security policies

While these shortcuts can help developers meet the DevOps imperative of delivering software faster, they also increase business risks. Ignoring security in DevOps ultimately increases costs by exposing the entire IT infrastructure to expensive data breaches, failed audits, application downtime, and advanced persistent threats (APTs).

[3] Venafi and Vanson Bourne. *2016 CIO Study Results: The Threat to Our Cybersecurity Foundation.* 2016.
[4] CSO Online. *As Containers Take Off, So Do Security Concerns.* September 17, 2015.

Provisioning new servers or virtual machines, checking code into and out of repositories and packaging application builds have all become automated. So anything slowing the process runs the risk of being shortchanged or completely ignored.

Automate Security as Part of DevOps

To overcome these issues, organizations must consider the entire process of procuring and provisioning keys and certificates as a fundamental aspect of the DevOps environment. The Venafi Trust Protection Platform helps organizations standardize and automate the process and insert these activities directly into their existing DevOps workflows. The Venafi API allows organizations to not only make use of their existing workflows and processes, it also provides the flexibility to integrate into any DevOps platform, such as Chef, Ansible, Puppet, Docker, HashiCorp, and more. Moreover, Venafi allows users to continue using the Certificate Authorities (CAs) of their choice.

The Venafi Trust Protection Platform is designed to work within existing environments as a fully-automated certificate service, so development teams can focus on delivering new applications and IT services quickly without security hampering their efforts. Venafi can accelerate the development process by reducing the time it takes to procure and provision certificates from days to minutes through automation.

Security teams can centrally define policies through the Venafi API and enable DevOps to properly comply with security policies and best practices by building in security from the beginning. The Venafi Trust Protection Platform provides the following benefits:

- Generates and issues unique keys and certificates on demand, usually in seconds
- Extends the certificate management platform used by security teams and system administrators to DevOps
- Provides a single view of security posture and compliance with integration to Help Desk systems and SIM/SIEM environments
- Automates certificate remediation and re-enrollment to align with policies
- Automates alerts based on certificate anomalies detected inside an organization and across the internet
- Delivers virtually infinite scalability without additional administrative overhead

Use Case—
Integration with Chef Framework

DevOps relies on standardization, automation, and close collaboration across teams to deliver IT services quickly, and developers expect the certificate process to fit into this paradigm. Provisioning new servers or virtual machines, checking code into and out of repositories, and packaging application builds have all become automated. In this fast-paced environment, anything slowing the process runs the risk of being shortchanged or completely ignored.

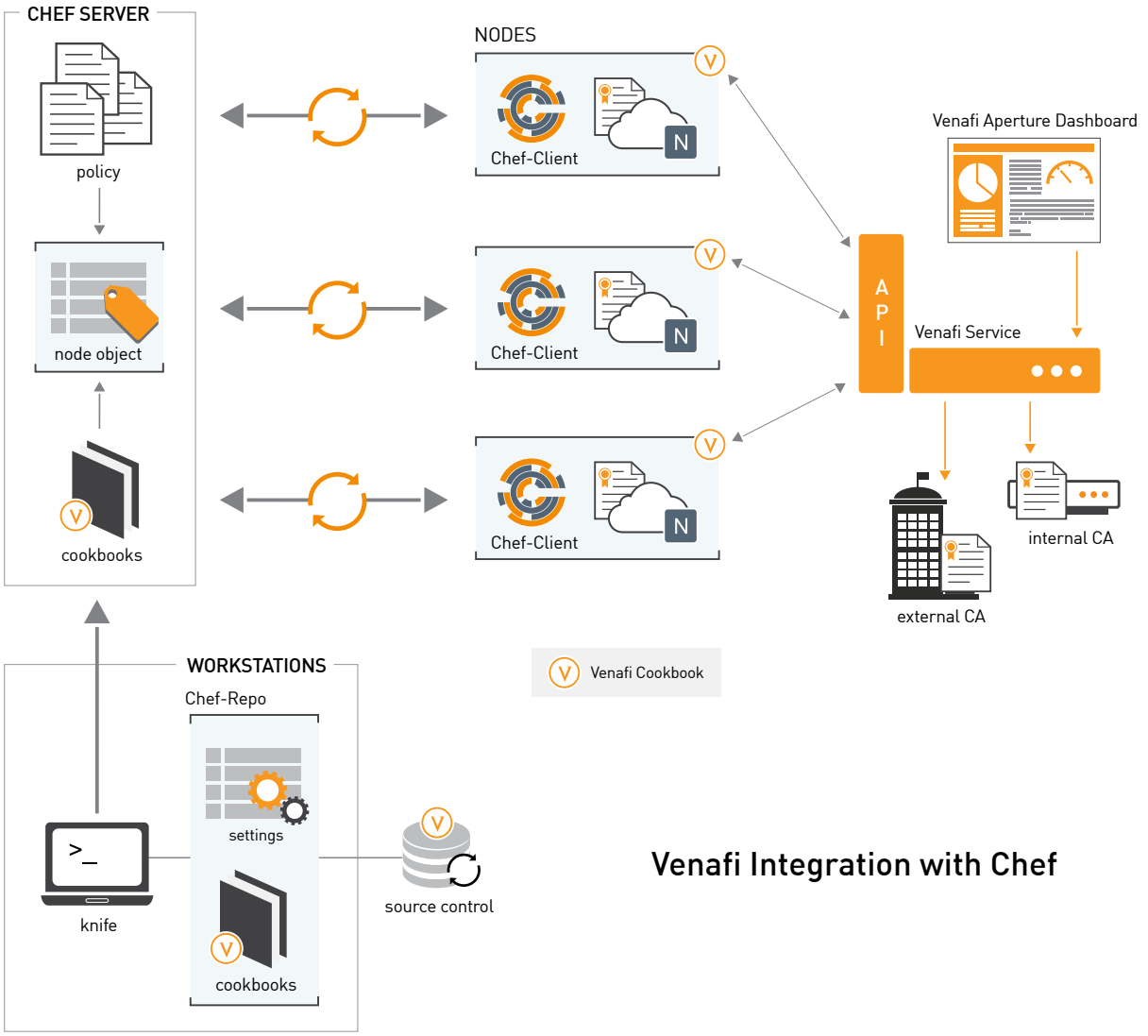
Venafi removes this traditional bottleneck in the DevOps process by integrating with DevOps platforms to automate the issuance of keys

Venafi can actually accelerate the development process by reducing the time it takes to procure and provision certificates from days to minutes through automation.

and certificates directly with the various DevOps platforms. Plus, Venafi provides sample cookbooks for Chef and other platforms that can be used to get started quickly.

The information below provides an example of how the Venafi API can be used to integrate the certificate process into a new or existing Chef framework.

1. A Venafi Cookbook is created and tested on the Chef-Repo. If required, it can then be synchronized to the source control system
2. The Chef Knife command-line tool can then be used to push the Venafi Cookbook to the Chef Server
3. The Chef-Client is used to access the recipes within the Venafi Cookbook. The Venafi Cookbook sample recipes currently include: request new certificate, check certificate status, retrieve certificate, and revoke certificate
4. The Venafi Cookbook recipes use HTTPS to connect, authenticate, and directly consume the available Venafi services (e.g., request, receive, revoke certificate)



Venafi Integration with Chef

Venafi Trust Protection Platform injects keys and certificates into Docker containers using the Docker events subsystem and API.

Use Case—
Integration with Docker Containers

For Docker, Venafi automatically injects keys and certificates into Docker containers as part of the container lifecycle.

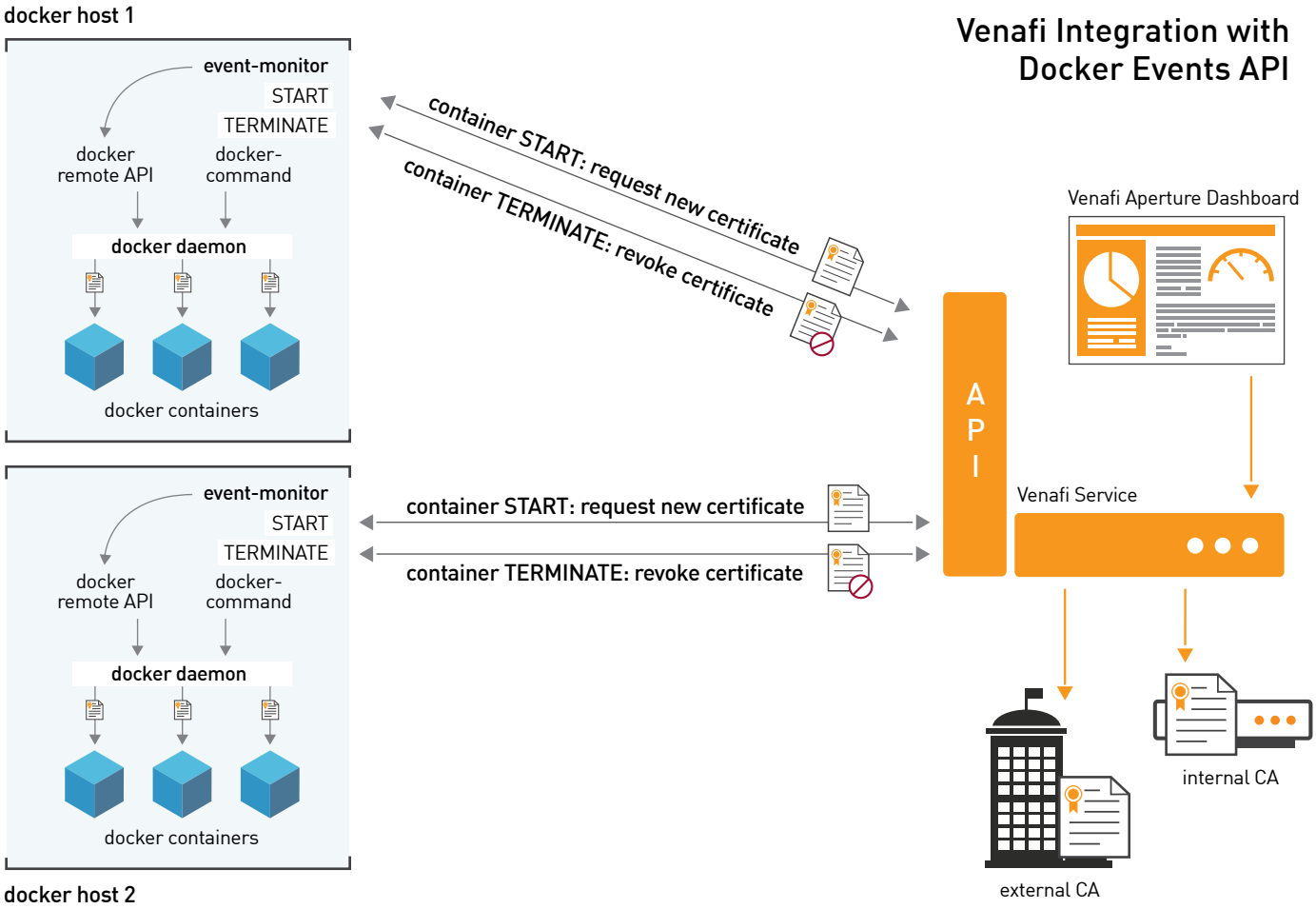
Microservices and container-based applications are often loosely coupled and not configured to communicate securely. In an environment that's moving so fast, it's imperative that applications are properly secured—like any other service with Transport Layer Security (TLS) authentication and encryption.

The example below shows at a high level how the Venafi Trust Protection Platform injects keys and certificates into Docker containers using the Docker events subsystem and API. A customer-provided daemon (background process) subscribes to the Docker event stream. For this simple example, the daemon registers for notifications of container "start" and "terminate" events. When a start event is detected, the following process is initiated:

1. The container name is retrieved
2. The Venafi API is used to lookup and retrieve the details of the pre-defined certificate policy
3. A new private key is generated and held in memory
4. A new certificate signing request (CSR) is created using the Venafi policy details retrieved in Step 2
5. The CSR is submitted to Venafi using the name retrieved in Step 1
6. The process waits briefly, then checks the certificate status with the Venafi platform.
7. When ready, the certificate is retrieved and combined with the private key currently held in memory
8. The key and certificate are archived and injected directly into the container using the Docker API
9. If required, the Docker Exec command is used to start or restart any services that depend on the key and certificate

Conversely, when a container is terminated, the following process is initiated:

1. The container name is retrieved
2. A certificate revocation request is made to the Venafi API
3. Venafi relays the revocation request to the Certificate Authority configured within the policy to revoke the certificate



ABOUT VENAFI

Venafi is the market-leading cybersecurity company that secures and protects keys and certificates so they can't be used by bad guys in attacks. Venafi provides the Immune System for the Internet™, constantly assessing which keys and certificates are trusted, protecting those that should be trusted, and fixing or blocking those that are not.

Conclusion

Automating security throughout the process can enable DevOps to maintain speed *and* ensure security throughout the environment. If your organization is still relying on manual processes for procuring and provisioning keys and certificates, chances are it's either not getting done or is being done poorly. The risk of inadequate security is exposing your organization to costly data breaches, critical application disruptions, data loss, and failed compliance audits.

But it doesn't have to be this way. The Venafi Trust Protection Platform can standardize and automate the issuance of keys and certificates within your existing DevOps environment—Venafi API integrates with configuration management and container platforms like Chef, Puppet, Ansible, Docker, and more. Stop sacrificing security: your DevOps teams can deliver IT services quickly and cheaply while adhering to your key and certificate policies.

NOTE: The use cases depicted here are intended to provide high-level examples of how Venafi APIs can be used by security teams to provide key and certificate services to DevOps teams. Since most of the components fall outside the Venafi domain, the solution has not been subjected to any security validation, screening, or ratification by Venafi.