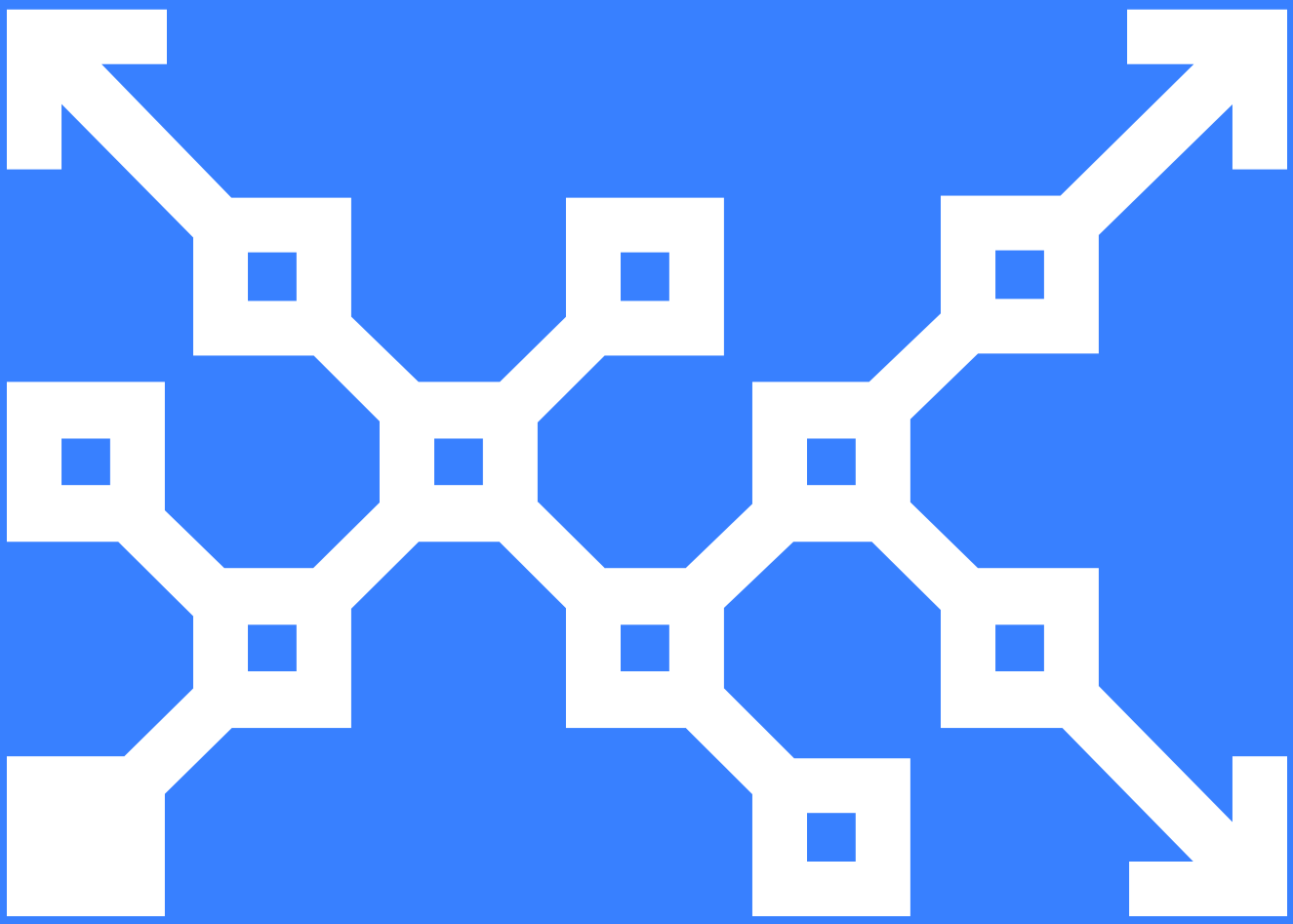
puppet

WHITE PAPER

Driving Value by Automating Compliance

Top 5 Questions from IT



It's never been easy being an IT professional, but over the past 20 years it's become particularly challenging, especially in large organizations. What had once been principally an IT sprawl problem quickly became a regulatory compliance nightmare. There are now more than a dozen major regulatory frameworks which vary by industry and geography — CIS, PCI/DSS, GDPR, NIST, HIPAA, to name a few.

These frameworks contain potentially hundreds of rules, and most companies need to adhere to up to 13 frameworks to be completely compliant in their markets.[1] Each of these rules represents a configuration setting on an individual piece of infrastructure. And there's lots of it — the typical IT estate has multiple technologies, vendors, OSes, applications, networks and assorted security infrastructure across the globe. Some are hosted on premises and some elsewhere. Many of these systems and applications are long in the tooth, but essential to the business. 50,000 nodes is not uncommon. Sound like fun?

But what if you could automate the compliance process? What if you had model-driven automation with systematic, continuous and prescriptive methods for defining your infrastructure and policies as code? Easier said than done, but clearly the only practical solution for managing this chaos. Gartner believes that integrating compliance as code into your DevOps toolchains (for starters) will improve your lead time by 20%.[2]

Gartner predicts that, by 2023, 60% of organizations in regulated verticals will have integrated compliance as code into their DevOps toolchains, improving their lead time by at least 20%.

At Puppet, we speak to lots of IT and compliance executives about automating compliance. Here's what we see as their top 5 questions as well as our perspective on moving forward with confidence.

1 "Compliance activities and fines cost organizations nearly $4m per year," *Security Magazine*, October 15, 2020

2 Betts, et al. "Innovation Insight for Continuous Compliance Automation," Gartner, August 11, 2020.

Question 1: What's the best approach for automating compliance and security in a complex global network? How do we begin?

First, you need to establish the scope of your compliance program. What are the regulatory frameworks that impact your organization and which are the most important?

Second, assess the full extent of your IT assets. What are the platforms, OSes, applications, networks, etc. Where are they located? Who manages them? Understand what specific assets you need to keep compliant against the regulatory frameworks you operate under. This will likely be a subset of your estate.

And third, prioritize which platforms would yield the most bang for the buck from automation.

Now you have a foundation and a plan. You're ready to begin. Our advice is to start small. Automate one thing across multiple servers, a single process, or everything on one or two servers within a small department. Start with a single benchmark, automate that and then move on to other benchmarks. Land and expand. This won't happen overnight, but it's worth it.

So why is automating compliance controls so hard? Funny enough, it's not the automation itself. Sure, there are lots of different OSes and platforms, but the real issue is cultural. Many people resist change. They have a fear of the unknown. Many staff have been doing configuration management manually for most of their careers and they feel that automating the process could result in a loss of control and unintended side effects. And they are right. But the flip side is that the risks can be managed and automation will free them up to focus instead on more strategic initiatives that drive innovation and growth.

There's also a challenge finding people with the right skillset and often, it's tough to train existing staff. People get used to doing things the same way and develop an ingrained mindset. There's often a conceptual challenge in learning a domain-specific language (DSL), understanding Git and learning how to do version control. You're asking people to adopt a change in their operating model and do some simple coding vs. running a command. Many people will embrace the opportunity to learn something new but this will not be universal.

Question 2: On a practical level, how do I automatically apply a common set of compliance and security controls across a wide variety of platforms (OSes and h/w), applications, compliance frameworks and geographies?

Managing the initial and ongoing configuration of a large network is a massive challenge. There are multiple platforms running multiple flavors of operating systems all configured with a different set of system commands. Traditionally, this has been handled by remote monitoring and management software (RMM), scripting and/or very strict change control. The problem is that these are really mitigation strategies vs. compliance solutions and none can scale in even medium-sized networks. For RMM and scripting, you need to deploy to every server and execute on every server. Not scalable. Not fun.

For deployed infrastructure, declarative, policy as code-based tools like Puppet can help you achieve the control and uniformity you need. They allow you to define a compliant configuration for any regulatory framework at a high level and then apply and automatically enforce that configuration on different devices across the estate every 30 minutes (configurable). (This approach owes a large debt to the early designs of Luke Kanies, Puppet's founder.)

The configuration is established in a centralized manifest along with the classification settings for each category of infrastructure. This makes it easy to determine whether the same configuration rule needs to be specified differently on different infrastructure. You don't have to tell it. No need to log into each server or understand the OS-specific commands and steps needed for configuration.

Question 3: How do I assess compliance posture across my estate in real time? How do I detect and remediate drift automatically?

Primarily, there are two ways of doing a compliance assessment. The first is using a remote network scanner. The scanner logs into every device individually and performs an assessment over the wire, over the network. The other is agent based, where you install an agent on each machine. The agent gathers the data locally for that machine and communicates the results back to a central server.

There are many advantages to using an agent-based approach because it's local. Being local means it's faster. And you don't have a network impact. You don't have to deal with things like credentials and login. It also means that you can scale a lot faster, because instead of having ten computers being scanned at one time by your network scanner, you can scan 50,000 at the same time by executing a single command across local agents. All of those scans will operate independently, with no impact on one another. And when they're done, they will send the results back to the central server so you're not consuming lots of network bandwidth. A traditional remote scan of a system might easily amount to gigabytes of traffic.

Another benefit to the agent model is its ability to automatically remediate drift. For example, the Puppet agent runs every 30 minutes (configurable). It checks into the central server and self-checks its configuration against the compliant manifest on the server. If it's out of sync with the compliant configuration, it nudges itself back into the compliant configuration. Compliance controls are enforced automatically every 30 minutes and all changes and remediations reported. Many of our customers say that once an auditor sees that they're using Puppet, that's all they need to know.

Question 4: How can I develop a policy-based approach to continuous compliance and provisioning on a global level?

The combination of declarative code (like Puppet) and a robust and granular classification capability make the specification and enforcement of policy as code straightforward. Assuming you have already done an assessment of your estate, it's a question of building a taxonomy with sufficient classification depth. It's about tiered logic and the profiling of assets.

For instance, looking at your network, what is the highest-level meaningful distinction? It's probably location. Under that may be data centers (on prem and cloud), followed by equipment type, OS, application type, etc. There may even be a value associated with business criticality (i.e., a development machine vs. customer-facing). These attributes may vary by organization, but the intent is that they tell the policy as code-based tool what the asset is in sufficient detail. The more granular the tool's classification capability, the more precise you can be in specifying conditional policy controls.

Then you look at the taxonomy and see which compliance frameworks map to which assets. Also, there may be other internal policies that are enforced through resource configuration that need to be addressed. In both instances, policy decisions are instantiated in configuration profiles in a centralized manifest. Once these are deployed, they can be automatically enforced and new asset profiles can be established. This approach is often referred to as policy as code and compliance checks can be embedded into DevSecOps environments to catch policy-related coding errors early in the software development lifecycle, prior to deployment.

Question 5: How do I overcome organizational silos between IT Ops, compliance and security? Old problem, but still isn't solved.

Again, many of the challenges associated with automating compliance have to do with people and not the technology itself. Most large companies perform automated vulnerability and compliance scans on a regular basis, either once or twice a year or quarterly. Scans are also done ad hoc in anticipation of an audit.

What normally happens is that Security/SecOps or Compliance will do the scan and then pass along a very large, difficult to read hard copy report to IT Ops. This is usually a PDF or Excel spreadsheet. It's then up to IT Ops to address all of the security and compliance failures in the report. The remediation process is manual and labor-intensive. (And the "over the transom" nature of the handoff doesn't inspire feelings of goodwill.) Once the fixes are made by IT Ops, they need to request another scan from Security or Compliance to see if the fixes worked. This can take some time and cause additional frustration.

The introduction of continuous compliance through automation has forced the conversation between SecOps, IT Ops and Compliance. Instead of a massive written report, automation gives you the ability to capture the scan results in embedded forms so the data can be moved into the policy as code tool (in our case Puppet) to perform the remediations. Once the mitigations are implemented and the desired state is confirmed with a follow-on scan, all future scans should come back clean (or close to it). Any out-of-spec configurations will automatically revert back to the desired state. Suddenly, everybody's job is easier.

But that's only part of the story. Policy as code-based tools like Puppet put everybody in control by removing bottlenecks. Teams can now run their own scans and control the complete development, validation and delivery of infrastructure as code to automate the services they own. IT Ops can run their own scans and perform remediation centrally. The results of the scan, instead of being a massive PDF or spreadsheet, show a clear mapping between the failure and the source of the fault on a specific server. Once the fault is corrected, IT Ops can run a scan to see if the change worked. Similarly, the security and compliance teams are in a position where they can run their own scans, as well. Now, things are running smoothly and different departments are working together. And once an organization reaches the "desired state," that configuration can be enforced automatically.

It's hard to overstate the importance for the business of continuous compliance through policy as code. It transforms a manual, error-prone, inefficient and clunky process that's a drag on operations into a highly efficient value creation engine that eliminates impedance and drives agility throughout the organization.

What to do next.

A successful configuration automation program is all about change management. At Puppet, we've been surprised at how often IT executives surface the cultural challenges associated with automation as something that caught them off guard. One way to address this is through careful planning, getting buy-in from the IT, Security and Compliance organizations and leveraging visible support from senior executives. We'd also recommend being very proactive in your communications with the staff and addressing any training needs well in advance of the rollout.

And as mentioned above, make sure you understand what you want from your compliance program and that you have a detailed and current picture of your estate. It will save you lots of time down the road.

Puppet is driving the movement to a world of unconstrained software change. Its revolutionary platform is the industry standard for automating the delivery and operation of the software that powers everything around us. More than 40,000 companies — including more than 75 percent of the Fortune 100 — use Puppet's open source and commercial solutions to adopt DevOps practices, achieve situational awareness and drive software change with confidence. Headquartered in Portland, Oregon, Puppet is a privately held company with more than 500 employees around the world. **Learn more at puppet.com**