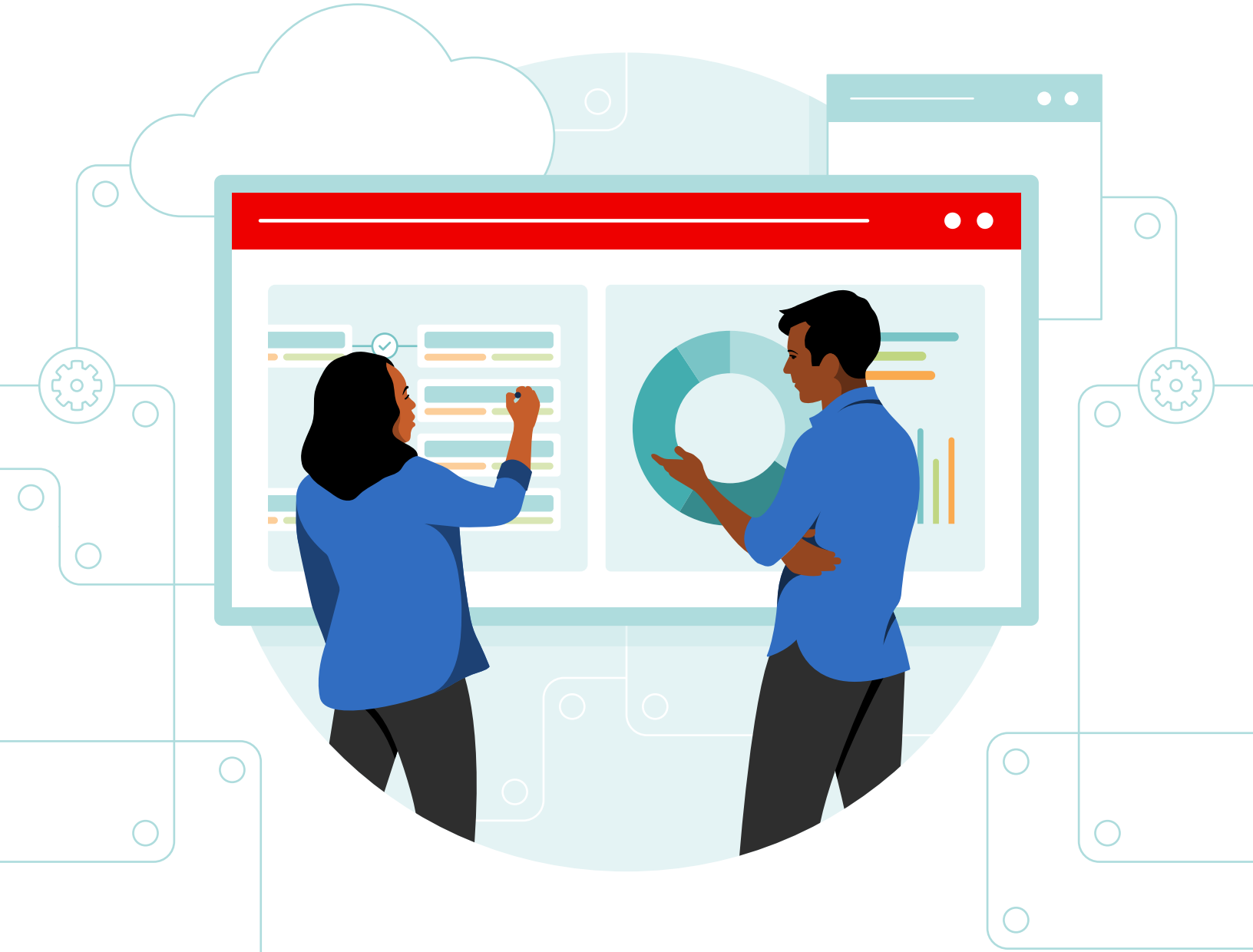
Red Hat

E-book

Tales from the field:

A system administrator's guide to IT automation



Introduction

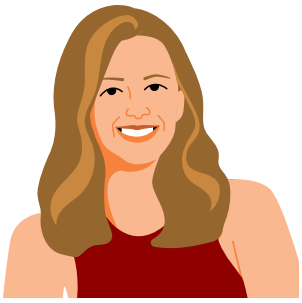
By Emma Van Sant, Principal Program Marketing Manager, Red Hat

Welcome to *Tales from the field: A system administrator's guide to IT automation*. This compilation of short stories seeks to share the excitement, frustrations, successes, and challenges associated with incorporating IT automation into organizations and teams across the globe.

Few technologies empower organizational transformation the way that IT automation can. IT automation improves security and compliance, abstracts away complex tasks for those with limited technical knowledge, improves standardization across the organization, helps organizations scale, improves continuous delivery, and reduces operational complexity and cost. But when adopting a holistic IT automation adoption mindset and approach, these business benefits come with an equal number of challenges.

At the most fundamental layer, automation is really about people. A popular phrase within our team at Red Hat is "technology is easy, people are hard." While this sentiment undoubtedly glosses over the technical difficulties associated with adopting a new technology, it's clear that the phrase has merit. A recent study of the role of enterprise-wide IT automation in digital transformation efforts indicate that a lack of skills, reluctance to change, and isolated, disconnected teams are all top barriers to widespread automation adoption.[1] People are the key to any successful technology implementation and adoption. Without buy-in, training, support, planning, and security, automation adoption initiatives won't be set up for success.

This book is a compilation of stories about these very challenges, written by Red Hat authors who have struggled with cultural, emotional, and practical barriers to holistic automation adoption within organizations. These writers have spent years either implementing automation within their organizations, or decades consulting on the implementation and adoption within other teams. This book is a good starting point as you become your organization's IT automation expert—and explore what that means for you and your career.



Emma Van Sant

Emma is a Principal Program Marketing Manager and Team Lead focused on Global IT Automation at Red Hat. She's spent the last 11 years working in IT organizations in the Czech Republic, Germany, and the U.S. and currently resides in California. She's passionate about communicating the value of leveraging open source solutions to address our most challenging IT problems.

1 Censuswide, sponsored by Red Hat. "Thriving through change with enterprise-wide IT automation." 30 Oct. 2023.

Chapter 1

Dispelling common IT automation myths

Why I was scared of IT automation

Learn the perspectives of 3 IT roles and what fears they might have around IT automation.

By Piyush Patel, Senior Managing Architect, Red Hat

Automation is no longer just a nice-to-have. Organizations are realizing that because of the nature of today's ever-changing IT world, it's crucial that they're able to scale quickly. To remain competitive and to be successful, they need to adopt an automation-first mindset (we'll talk about how to adopt that mindset in chapter 3).

As with any transformative technology, the adoption of IT automation can spark worry among those tasked with implementing it. In our experience working with customers, we've found that 3 roles in particular are mostly likely to have concerns about IT automation. In this section, I'll explain what's behind these common fears and how to address them.

The executives

IT executives are usually the most directly responsible for budget decisions. When it comes to automation, they tend to be most concerned about how to prove the return on investment (ROI) of an organization-wide automation strategy.

Executive confidence is essential for an enterprise automation strategy to be effective; organizational support needs to come from the top down. Fortunately, the concerns of IT executives are fairly simple to address.

Because IT automation addresses specific, manual tasks or processes, calculating an accurate cost or time to value is much less complex than estimating the ROI of other software solutions for IT operations. By breaking down a larger process into smaller elements to be automated in a step-by-step manner, you can identify incremental results as you progress and then the larger, aggregate ROI once the entire process has been automated.

IT managers

IT managers are much closer than IT executives to the processes that will be automated. They're managing the teams that are currently performing the work that IT automation

can address. While IT executives are concerned with how they'll be able to prove ROI, IT managers are concerned with what that ROI is. They want to make sure that the investment in an automation solution will help them get the most out of their teams, and not simply be another technology to manage.

The goal of IT automation is to free up top talent to focus more on proactively delivering value, and move them out of fire-fighting mode. To address uncertainty about what value automation will offer, IT managers should evaluate their teams closely to identify which mundane, repetitive, day-to-day tasks are demanding time and attention from highly skilled members. By pinpointing these tasks, IT managers can understand where automation can deliver the greatest benefit.

Reframing these concerns as opportunities can be a helpful way to accept change and move forward with confidence.

Look for manual processes that serve end customers, such as service ticketing—could they be automated to provide self-service capabilities? Or what about applying automation to the tasks that most often lead to those service tickets? Automating infrastructure management through practices like Infrastructure as Code (IaC) can streamline processes that ensure consistency and reduce manual effort.

Knowledge is power. The more IT managers understand about their team and the routine processes occupying their time, the more confident they'll feel in the value that automation can offer.

Sysadmins and engineers

Sysadmins and engineers have much more personal fears about adopting automation. They're closest to the processes that will be most affected by the implementation of an automation solution—they're performing the very manual tasks that automation will address. Common concerns of these roles include *Why do I have to learn something new? Will this actually make me more efficient? Will it replace me entirely?*

Those questions aren't unfounded. Moving away from established, familiar processes can be stressful. A significant reorganization of a person's workload can feel alarming. But without buy-in from these roles, an automation strategy is likely to fail. After all, these are the people executing the automation strategy, which in turn proves the ROI of an automation solution.

Reframing these concerns as opportunities can be a helpful way to accept change and move forward with confidence.

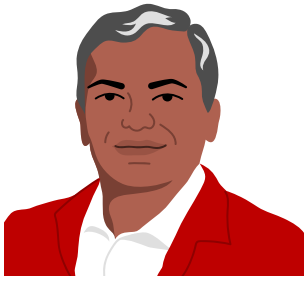
IT automation isn't just about unlocking greater organizational efficiency. For sysadmins and engineers, IT automation can help free them from mundane workloads and give them an opportunity to be productive in new, deeper ways. And not just more productive, but influential. These roles can be leaders in helping promote an enterprise-wide automation strategy, innovating with automation, and delivering greater, more visible business value.

Enterprise adoption of IT automation strategies is accelerating. Automation skills can be a key differentiator in the job market. Training, such as Red Hat® Training and Certification, can not only help better prepare sysadmins and engineers to effectively work with automation and embrace their evolved role, but also to be more marketable.

Conclusion

Whether or not an enterprise automation strategy is successful depends hugely on the support of key internal stakeholders. Understanding and addressing their automation concerns is essential.

→ **Read the 5 steps to automate your business e-book**



Piyush Patel

Piyush is a Senior Managing Architect in Red Hat's Ansible Automation Platform Practice. He joined Red Hat in 2017, working with the global telecommunications team before moving to automation. He is passionate about automation and has a strong background in all aspects of datacenter, virtualization, networking, and cloud. His motto is: "If you are going to perform a task more than twice, then spend time to automate it."

4 IT automation myths dispelled

Identify and counter common IT automation myths.

By James Mighion, Principal Site Reliability Engineer, Ansible Cloud Services

In my career as a DevOps engineer, I have spent hundreds of hours automating numerous mundane tasks. Whether you are just beginning to use automation or already have some experience, you may run into resistance due to many common IT automation myths. I'd like to address some of these myths based on my experience.

Myth 1: Automating a task takes more time and effort than it's worth

If it takes more time to automate a certain task than simply to accomplish the job manually, it is not worth automating.

You are likely to get resistance from your peers or management about automating tasks based on time savings. In reality, every job you do as an engineer is worth automating, but you have to be cognizant of the time and deliverables. When certain tasks appear to be not worth automating, I've often found that what it really means is that it's just not possible to automate those tasks at this time. Your objective should be to automate the task in the future—you are likely to get less resistance from your team if you keep this perspective. Just make sure to communicate the automation proposal in a way that meets your immediate goals and improves future effectiveness.

Remember that this process can and should be iterative. You don't need to tackle everything all at once. Start with something small and come back to it later to make it even better. That way, you'll achieve some benefits while making it easier to improve upon next time.

Myth 2: You don't need to automate a single occurrence task

I only need to do this once, why should I bother automating it?

This is probably the biggest myth I have seen during my career.

Here's a real-life example: A product customer raised a specific issue, and it needed a particularly complex setup. The person working on the task asked me and my fellow DevOps team members for help. We did our best to automate the task we were asked to do. The

task owner did his verification and then asked if we could keep it up and running for a little while. And that "little while" quickly turned from days to weeks to months. (Development environments often become production environments, so keep that in mind when you're asked to create a "quick" setup or fix.)

We then received another request to tweak a virtual machine (VM) instance, then a follow-up request to create a snapshot, and eventually a clone of the VM. Once the VM had been cleaned up by automation, they needed to set it up all over again.

Every time a new request came up, we kept asking the person to automate it, but we were met with resistance from both the individual and management because they considered it a single occurrence task. That was technically true, but it required a lot more effort than that. Had they approved more time for automating the task up front, they could have saved a lot of time—for themselves and us. Every time they needed to make a change, they could have noted it as part of an update to their automation and made everyone's lives easier.

This is just a single such example. While it is sometimes hard to know how much work will be involved in solving a problem in the beginning of a project, it is essential to step back at an appropriate time and evaluate whether you need to invest in automation—before it's too late.

If you can do it manually, it can be automated.

Myth 3: Automation breaks, so don't waste time doing it

It is not worth investing time in maintaining automation because it breaks often.

It's true that automation breaks periodically as the various scripting languages change over time—or the system you interact with through the automation deprecates or introduces features.

But does that make your automation effort worthless? Absolutely not. I have personally experienced this when we used automation to build Red Hat Virtualization environments. The automated tasks were initially designed for version 4.2, but we were soon upgrading to build on the 4.3 and 4.4 versions.

We kept tuning and tweaking our automation and handling the various scenarios and quirks of each version. The results were a versatile combination of Red Hat Virtualization deployment automations that acted as engines that, when fueled with appropriate inputs for each of the versions, produced fully deployed and configured Red Hat Virtualization environments.

Did I spend time debugging and fixing the automation? Yes. But I definitely found the effort worth it every time I had to rebuild the Red Hat Virtualization environments. We had 6 to 7 environments, each with its own version, sizes, and other characteristics. Whenever new builds became available or an environment became stale due to abuse during testing, it just took a single button click to reprovision the resource.

Automation maintenance empowers you with confidence. While maintaining automation is time-consuming, in my experience, it has been more effective than I initially thought. And all things considered, what IT work doesn't include maintenance?

Myth 4: It is impossible to automate this task

It is tough to automate this. It just can't be done.

There are times when you are faced with specific processes that are more difficult to automate than you'd hoped. It is not uncommon for you to hear from your peers (or read on the internet) that a given task is just too tough to automate—nobody has ever done it, and it probably can't be done.

I ran into that situation early in my career when I was automating various infrastructure tasks. I lacked experience, and others told me it couldn't be done. My boss at the time still wanted me to pursue the challenge. His willingness to give me more time on the problem, while acknowledging that it was tough, gave me additional motivation.

After spending about a month exploring options and trying various methods to create proofs of concept, I saw that I could "semiautomate" the task under the given constraints of the situation, meaning there were a few manual interventions. So it was indeed tough, but it was still a myth that it just couldn't be done. It takes determination and courage to go after a tricky automation problem.

Here's another perspective: If it has never been done, you invent something new. And that's what I did. The more experienced people were right: It couldn't be automated. However, what they really meant was that it couldn't be fully automated. And what are engineers if not creative problem solvers?

If you can do it manually, it can be automated.

Conclusion

People tend to believe automation myths. It is important to understand that automation can be time-consuming, but it is likely that it will be of value to you and your team. There will be times when automation breaks and fixes need to be made, but it will pay for itself by saving time and reinforcing the benefits it provides.

Sometimes it is difficult to automate tasks, but take those challenges as an opportunity to innovate and share the knowledge you gain. Lastly, it is very likely that when you do a certain task once, you will have to do it again, and if you automate it, you set yourself up for success.

James Mighion

James is a Principal Site Reliability Engineer in the Ansible Cloud Services organization. He's been with Red Hat since 2011 and has had many different roles along the way. James is passionate about automation and contributing to open source projects.

How to automate routine tasks to avoid mistakes—and save time

By Thomas Tuffin, Senior Software Quality Engineer, and James Mighion, Principal Site Reliability Engineer, Ansible Cloud Service

The job responsibilities of engineers, sysadmins, and DevOps roles include answering dozens of emails, upgrading hundreds of servers, and patching all the systems in your datacenter or your favorite cloud. All of these tasks take a considerable amount of time to complete—and they are sometimes boring but always important. A slight mistake can open up a potential threat to the security of the entire system. For example, failing to apply a security patch can make a system vulnerable. And let's be honest, people make mistakes. But what if you could automate these mundane tasks, save some time, and simplify your team's tasks?

What tasks should be automated?

Repeatable and routine tasks

These are the tasks that need to be carried out on a regular basis. For example, we need to update my GitHub fork, including the latest changes from an upstream repository. Other examples include:

- Collecting the system backup logs.
- Sending out a weekly system upgrade email.
- Carrying out upgrades.

Simple, yet time-consuming tasks

You might be required to collect system utilization in your lab or send a daily email to the team about system availability. There will be some tasks that take a lot of time for the system to complete, such as a continuous integration/continuous delivery (CI/CD) pipeline or a particular installation. It's always better to set some alerts in such cases. We use an email alert that we receive after each Jenkins pipeline completion. This way, we can keep track of progress and also use the time to do other tasks.

Complex tasks

The combination of several jobs, such as installing an operating system (OS), installing the latest packages, and making sure a weekly CI/CD pipeline is run into the system, is not complex, but think of repeating these steps for hundreds of systems in your datacenter or cloud environment. Other examples include setting up load balancers and proxy servers, or making entries for systems in your Domain Name System (DNS) infrastructure.

Automating virtual machines

One of the points of automating tasks is to get the mundane, repeatable tasks scripted and deployable. Once you accomplish this, it frees up your time to work on more high value or strategic, proactive tasks for the team.

One example of a mundane task to automate is creating VM templates out of new builds of Red Hat Enterprise Linux®. Let's say we were using Red Hat Enterprise Linux 9-based VM templates, and then Red Hat Enterprise Linux 10 was released. We need to have all our templates updated with the new version. That would be a painful and error-prone task to perform by hand.

Another mundane task is provisioning VMs requested by your team members. Automation makes such tasks much more efficient. Even better, create a self-service portal that your team can use to request new VMs. If the VM has a unique configuration, you'd intercept that request and take appropriate action. Otherwise, let the automation handle VM creation and send out a notification (email or chat message) when the VM is ready.

Another important aspect people forget about is removing VMs to preserve resources. Engineers often forget to clean up their resources in a timely manner. As an admin, you could keep manually deleting VMs after asking if anyone is still using them, or just create a set of rules for the longevity of VMs. When the VM fails to comply with those rules, it is automatically deleted.

It's never about automating yourself out of a job—it's about deciding when you have done enough of the work and want to change your focus.

Conclusion

We have given various examples of simple, repeatable tasks that are often time consuming. If properly planned, and with the help of some tools, these tasks can be completely automated and stored in a centralized repository like GitHub. By automating these steps, we can save a lot of time and increase team productivity. Most importantly, we can avoid making mistakes when a complex task is involved, which usually depends on taking input from 1 segment of the system and passing it on to another segment, and then moving on to completion.

It's not always necessary to automate every job. If you feel automation is difficult, you can usually start by working with small, low-risk tasks and then build up your repository by partially automating other tasks. Automation will bring a huge change to the team's work style in the long run.

In summary, it's never about automating yourself out of a job—it's about deciding when you have done enough of the work and want to change your focus.

Thomas Tuffin

Thomas is a Senior Software Quality Engineer for Red Hat, working on the Event-Driven Ansible component of Ansible Automation Platform. He is passionate about open source and has a keen interest in emerging technologies that provide equal and fair opportunities to everyone. Always looking to discover something (or somewhere) new, Thomas prefers to either be traveling or working on a project.

James Mighion

James is a Principal Site Reliability Engineer in the Ansible Cloud Services organization. He's been with Red Hat since 2011 and has had many different roles along the way. James is passionate about automation and contributing to open source projects.

Chapter 2

The benefits of standardizing on an automation platform

Numbers talk: Getting your team on board with IT automation

See how you can automate rote tasks and shift your focus to more attractive projects.

By Andrius Benokraitis, Senior Manager, Technical Marketing, Red Hat Ansible Automation Platform, and Cindy Russell, Product Marketing Manager for Red Hat Ansible Automation Platform

Automation enthusiasts are often born out of the results that teams achieve through successful automation projects. For many of us, "sleep more" is perhaps the most compelling benefit when updates can be done consistently, correctly, and rapidly across hundreds or more systems, networks, and cloud implementations. Midnight calls to resolve an issue aren't fun for anyone.

And when it comes to security and compliance needs, we have seen teams face pressures to resolve risks quickly. When you can create the proper fix in an automation playbook, it is then consistently executed across all affected solutions in the global infrastructure. This can be fast and effective, so the risk is remediated quickly, without the human error that can occasionally creep in when you are doing the same things repeatedly.

But there are other benefits, too, such as allowing for more time to work on key priorities that add value or the ability to advance experience and skills. Here are some statistics we've used when trying to get some of our teammates onboard with IT automation.

By the numbers stories

The benefits of automation can be dramatic. There are numerous cases where automation has resulted in significant, measurable savings of time and effort when the right automation solution is being employed. Let's consider a few of them.

Accelerating configuration process time from 8 to 12 months to 6 weeks

Southwest Airlines found that building, testing, and deploying network access control (NAC) configurations—critical for preventing rogue external devices from connecting to Southwest's network—was taking 8 months or longer. With Ansible Automation Platform, the same process took only 6 weeks. Not only did Ansible Automation Platform help accelerate the process, it also helped Southwest enhance its response time in the case of critical system failures. Unlike engineer teams that need to manually check every network

layer, Ansible Automation Platform can quickly assess the network environment and identify issues to minimize downtime.[2]

Cutting deployment times from 20 minutes to 10 minutes

The UK's Department for Work and Pensions (DWP), which administers pension, working-age, disability, and ill health benefits, adopted a deployment orchestration tool in 2016 to support its on-premise environments. It was far from perfect, requiring separate codebases for different environments and needing to be run several times for each environment deployment, which took about 50 minutes. With Ansible Automation Platform, DWP no longer had to run several iterations, plus they can use a single playbook for all environments. By adopting Ansible Automation Platform, they were able to reduce deployment time from 50 minutes to 10.[3]

Cutting server set-up times by 90%

Dai-ichi Life Insurance Company, Limited (Dai-ichi Life), among the largest life insurers in Japan, has more than 1,000 servers in its IT infrastructure. In 2020, the company decided to implement automation to meet demands for high agility and efficiency in its server infrastructure and accelerate digital transformation efforts, and selected Ansible Automation Platform as its automation configuration management tool. By applying automation to its server set-up processes, Dai-ichi Life has been able to ensure quality while redirecting budgets to other IT investments, freeing up team members to work on other projects, and cutting the server delivery time from 5 weeks to 3 days.[4]

When IT must respond rapidly to needs like these, automation makes the job fast and efficient for you. You can find more stories like these on **our case study page.**

2 Red Hat case study. "Southwest Airlines is expanding its automation use cases." 2 May 2024.
3 Red Hat case study. "DWP advances automation with Ansible Automation Platform." 29 Oct. 2024.
4 Red Hat case study. "Ansible supports infrastructure automation at Dai-ichi Life." 22 Aug. 2024.

Automation skills in demand

As emerging technologies become more widely adopted, organizations need teams with relevant skill sets to deploy and manage them. Automation is no exception.

Manual processes or IT operations are seen as 1 of the top 3 barriers to digital transformation, according to Red Hat's 2024 Global Tech Outlook.[5] Automation can be a leading tool in helping organizations overcome that barrier. That's why automation, along with AI, is at the forefront of enterprise technology strategies today.

Developing Ansible skills can help prepare you for what's next in IT operations, and position you as an automation leader within your organization. In chapter 8, we'll explore the 8 skills you need to be successful in IT automation.

Deliver projects that matter

To best support the business, we believe that cross-functional IT teams must come together to deliver on important initiatives, such as digital transformation applications, expanded hybrid cloud platforms, modernized or migrated technology stacks, edge or Internet of Things (IoT) implementations, and more. Often these initiatives will involve learning or using emerging technologies, and they help demonstrate and deliver value to key stakeholders like customers and your company's leaders. When you have more time to focus on these advanced projects, professional development and growth can follow.

Automation can help you automate rote tasks and complete lower-value work, so you can instead focus on priority projects and innovations. For example, let's say you have to apply a patch to 500 systems to remediate a security vulnerability. Automation will dramatically reduce the time required, as we see in the customer stories above. As a result, you have much more time to focus on other important projects, which is good for both you and your company.

5 Red Hat overview. "2024 Global Tech Trends," 20 Feb. 2024.

With freedom from manual tasks and plenty of automation skills, you may be interested in advancing your career as an automation architect. This handbook will help you learn more about the best practices for forming an "automation-first" community and leading your company's automation strategy.

We hope you can see the automation benefits for you and your company. We wish you the best in your learning journey, and we hope to see you as an automation enthusiast at AnsibleFest at Red Hat Summit.

Recommended learning resources:

- Red Hat training module (free)
- On-demand and upcoming webinars
- Ansible Automation Platform videos
- The automation architect's handbook
- A free trial where you can build and run your initial automation projects

Andrius Benokraitis

Andrius is a Senior Manager in Technical Marketing for Red Hat Ansible Automation Platform. He brings more than 20 years in the computer software industry from companies such as IBM, Nortel, and Cumulus Networks. Andrius is skilled in network automation, enterprise Linux, business analytics, technical writing, and strategic alliances.

Cindy Russell

Cindy is a Product Marketing Manager for Red Hat Ansible Automation Platform. She brings many years of experience in developing technical practitioner and other marketing programs for software products, including automation, analytics tools, database and related machine learning technologies, and developer tools. She has worked at companies including IBM, Sun Microsystems, and several startup and smaller independent software vendor companies.

Use AI-enabled automation to combat your increased workload

Tired of mundane, tedious, boring tasks? Automation improves your efficiency and frees your time to focus on new and innovative opportunities.

By Ricardo Gerardi, Principal Consultant, Red Hat

As the information technology footprint grows in many corporations, so too does the role of a sysadmin. A modern sysadmin deals not only with physical machines and operating systems, but also with myriad virtual systems, cloud environments, network devices, and container workloads. The complexity and number of systems managed, even in small companies, is typically of a much greater magnitude than a few years ago.

Your role as a sysadmin consists of deploying, maintaining, updating, and ensuring that all these heterogeneous systems work correctly, guaranteeing that both customers and the business benefit from these technologies. No doubt, this is a dynamic job that provides plenty of opportunities for performing tasks that are challenging and exciting, and learning new skills.

However, due to the nature of the job and the number of systems you manage, you also, typically, have to perform several mundane tasks. Examples include applying a minor update on a group of servers or cleaning up some logs. You may be asked to report on installed software versions in preparation for an upgrade or in response to a security audit. These tasks, while simple, still take a long time to complete when combined with the number and variety of devices on which you need to perform them. Spending time on these tasks takes your time away from working on more valuable or more exciting activities.

While working on repetitive and boring tasks, have you ever experienced a low sense of fulfillment or felt overwhelmed? If you have, you're not alone. I've heard that from colleagues throughout the years, and I have felt it myself.

Making a change

So what can you do to improve the situation and make better use of your time? The answer can be complex and, in some cases, may require the company to address it systemically. For other problems, you can use your technical expertise to improve the way you work by automating these repetitive mundane tasks.

When we think of automating an IT process, it's normal to evaluate spending the time and effort to develop automation artifacts to address large or complex issues. There are many benefits to doing this. You can realize the same advantages by applying automation to simple and repetitive tasks, especially when you can reuse the automation to manage hundreds or thousands of systems.

Some of these benefits include:

1. **Efficiency:** Execute tasks more efficiently across a large number of targets, allowing the computer to do the hard work for you.
2. **Standardization:** Run tasks in a consistent way, ensuring the same results every time. For example, avoid common, distraction-led mistakes where you configure 1 device with lowercase characters, and another with an uppercase character, leading to a hard-to-troubleshoot issue later on.
3. **Less error:** Fewer people touching the systems lowers the probability of configuration mistakes or accidents leading to outages. For example, have you ever missed the WHERE clause in a DELETE or UPDATE SQL query?
4. **Personal fulfillment:** Working on more fulfilling and intellectually challenging tasks typically leads to higher job satisfaction.

How do I start?

Applying automation concepts to simpler tasks may be a good way to get you started with automation. You can see immediate benefits by tackling quick wins while learning and preparing to automate more complex issues later. In fact, that's how I started with Ansible Automation Platform a few years ago. I used Ansible Automation Platform to replace some scripts that ran periodic checks on a few hundred servers that I managed. I spent a few hours doing the initial configuration, and after that, I could use the same environment to automate these checks and many other small tasks, performing them quickly and consistently.

Applying automation concepts to simpler tasks may also be a good way to get you started with automation.

In my opinion, this is a good strategy for starting your automation journey. Look for an issue or task that you perform regularly, takes much of your time, or is annoying to do, and then select your automation solution and apply it to solve this issue while learning the inner workings of the solution and automation as a whole.

If you don't know where to start, here are some ideas of tasks that are good candidates for your initial automation project:

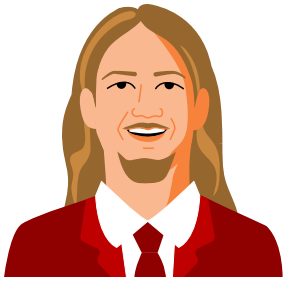
- Connectivity tests
- Packaging maintenance: Install/update/delete packages
- Reporting installed application versions
- Managing users
- Resetting user passwords
- Cleaning up unused files such as core files and logs
- Executing a backup or restore of a system or configuration

Use AI to accelerate automation adoption

You can also use AI-enabled tools to help you get started on your automation journey. Large language models (LLMs) specifically tailored for automation can help translate your sysadmin experience into automation, and accelerate adoption across your organization. AI tools can also help with testing, validating, and documenting your automation artifacts.

Where to go from here?

By automating simple and repetitive tasks, you can complete them more efficiently, freeing up some of your time to work on more important, or perhaps, more exciting, projects. The automation journey may be challenging, but it's definitely a journey that I recommend. Give it a try, and you may collect many benefits for your business and for you personally.



Ricardo Gerardi

Ricardo is a Principal Consultant at Red Hat, specializing in IT automation with Red Hat Ansible Automation Platform and Red Hat OpenShift. He has experience in the telecommunications sector, having worked as Senior Architect at TELUS, and had previous experience as Senior Consultant and Pre-Sales Specialist for Network Management solutions at IBM Brazil and IBM Canada.

Take control of your operations: Forge a new way to work

A case study about developing tools that automate and streamline mundane tasks to improve delivery time, reduce human error, and free up more time for new work.

By Thomas Tuffin, Senior Software Quality Engineer, Red Hat

You are a system administrator in an operations team that works using agile methodologies, following a framework based on scrum, kanban, or even scrumban. Your team takes care of a constantly evolving environment. On top of the daily support tickets, there are deliverables to stakeholders that must be completed according to deadlines. The workload is demanding, and the backlog is only getting longer. This situation creates a high-pressure environment where tasks that deliver immediate value to stakeholders are considered top priority and are given the most attention. Tasks that do not provide immediate value are given a lower priority and fall further down the backlog. Unfortunately, it is all too common for the development of automation tools and frameworks to fall into the low priority basket, and so your team continues to do things manually. If some of this sounds familiar, you are not alone.

Redirecting focus to developing automation is not always a straightforward matter. Hopefully, you will receive strong support from your team, and maybe positive reinforcement from management. However, you may hit some heavy resistance when it comes to prioritization—and for good reason. Your team has stakeholders to deliver to, and your primary mandate is to ensure they get what they need on time.

As an operations team, you likely have dependencies on other teams and suppliers. One delay up the chain will have a domino effect further down, thus delaying your deliveries. For teams working scrum, it is not uncommon for stories and subtasks that were not completed by the end of the sprint to spill over into future sprints. This backlog increases the pressure on your team, delaying future projects and pushing back the tasks that are primarily focused on improving your environment. Such a situation can leave a team feeling as though they have lost control, and they may feel frustrated as little attention is given to long-term solutions. Again, this is a common situation for many operations teams.

How to get started with automation

So, what can you do about it? You should, of course, begin by raising the topic with your team, product owner, or project manager, and, if necessary, your manager. Providing examples of how much time and effort will be saved by investing resources into developing

automation is a good way to get management's attention. You can begin by drafting a proposal that outlines how long tasks take the team now, how much time you think it will take to automate these functions, and how much time tasks will take after automation. You can also predict and measure time savings and ROI through the use of automation analytics.

Start allocating some extra time for those tasks that can be automated. Begin with the smaller jobs, such as streamlining the process of adding items to your configuration management database (CMDB). A good CMDB will have an application programming interface (API), so creating a shell or Python script to take advantage of that API is an integral part of the automated workflow for deploying and managing servers. Once you have the API figured out and a script that interfaces with it, you could take it a step further and build a library that other scripts can use. A library will standardize the way your team interacts with the API. It will also prevent duplicate development effort since all future scripts can use this library whenever they need to access the API. It may be a little bit more work up front, but it will save time and effort in the future development of scripts and tools.

Armed with a library with standardized access to the CMDB API and a script that can pull asset information, you can begin automating parts of your server deployment process. If you already use a preboot execution environment (PXE) and kickstart files as part of your deployment procedure, why not automate most of the work? You could achieve this through scripts written in a language of your choice or by using Ansible Playbooks. By employing some of Ansible's built-in modules, such as the template module, you can easily generate PXE boot files and kickstart files with data pulled from the CMDB. Ansible also has modules available for connecting to various manufacturer's out-of-band management interfaces, so tasks like setting out-of-band hostnames, IP addresses, and power management are straightforward.

Use the skills and tools at your disposal to streamline processes and reduce time spent on repetitive tasks.

You can, of course, achieve the same result through other tools, custom scripts, and various open source and proprietary software solutions. There are many ways to approach the challenge. The same is certainly true for server deployment. Use the skills and tools at your disposal to streamline processes and reduce time spent on repetitive tasks.

→ Explore more automation use cases

With a few of the smaller tasks now automated, you should start to see some positive effects. For instance, you and your team no longer need to click through that clunky CMDB interface to input and retrieve information. In addition, part of your server deployment process is now automated. Finally, there is another big benefit of automation—a reduction in

issues due to human error. It is well known in the IT industry that human error is a frequent cause of various system failures, outages, and security breaches. Automating even the most simple tasks, especially the mundane ones, will help to alleviate this.

Taking back control of your environment

As more of your team starts to get involved in the development of automation tools, you will find that the ecosystem of tools grows organically. That ecosystem may include AIOps capabilities, which augments what you do with automation and helps enhance your automated workflows. This growth will raise other challenges, such as code quality, maintainability, and responsibility. It is important to establish guidelines early on to ensure certain criteria are met and ways of working are followed. You can achieve part of this by using a version control system and tools such as Git and Gerrit to track changes and support code collaboration within your team.

Once you have put together a healthy toolset to take care of single, repetitive tasks, you can level up your automation by using **Event-Driven Ansible** to automate cross-domain workflows that deliver greater benefits across the organization. These workflows could include end-to-end incident response, or compliance audits.

After you've successfully demonstrated the value of automating both single task and cross-domain workflows, you will find that your team defaults to using automation. One of the initial questions they most likely will ask when planning something new will be "Can we automate this?" The more automation you implement, the more likely you are to change the status quo and take back control of your environment. Developing tools that automate and streamline mundane tasks will not only improve delivery time to stakeholders and reduce human error, but it will also allow you and your team more time to focus on working on the future of your environment.

Thomas Tuffin

Thomas is a Senior Software Quality Engineer for Red Hat, working on the Event-Driven Ansible component of Ansible Automation Platform. He is passionate about open source and has a keen interest in emerging technologies that provide equal and fair opportunities to everyone. Always looking to discover something (or somewhere) new, Thomas prefers to either be traveling or working on a project.

Case study: Save time and unlock new opportunities with IT automation

When you start imagining what automation can do for your business, you get more than just time savings—you shift your culture and open up new opportunities.

By Jered Cortez, Ansible Ecosystem Sales Leader, North America

Engineering teams getting calls at 2 a.m. on Saturday night to resolve a network issue. Support specialists waiting for admin rights approval to resolve a ticket. Executives feeling pressured to do more to deliver greater value, and in less time.

These are the kinds of scenarios I hear about often in the field, and they have customers wondering if automation could help them transform their IT operations to do more, better, and in less time, in order to meet and exceed expectations. The answer? Yes. Let me share a specific example of what that 'yes' looked like in practice for a particular customer.

In 2022, I was working as an enterprise sales specialist, focusing on helping customers discover the value of automation for IT operations and successfully adopting Ansible Automation Platform. One of those customers was a large beauty retailer in the United States.

The company sells over 25,000 products across more than 1,000 stores and their e-commerce website. They also offer a robust loyalty rewards program to support the customer experience. That's a lot of point of sale systems, a lot of platforms, and a lot of services—in other words, pretty complex IT operations. How do you keep that up and running reliably, while improving customer experiences?

Their journey with automation started with a single person. At the time, he was an engineer but he's since progressed into an automation architect. He was someone who saw the value of automation for himself, his team, and ultimately his company, but was more or less doing it on his own. He became the go-to person within the company, but we all know that creates bottlenecks; a single automation champion can only handle so much demand.

Largely because of the work this person was doing, the company saw an opportunity to expand automation use cases to help deploy in less time without disrupting its day-to-day business operations, and decided to expand that single automation champion into an automation team. They started to look for different ways to use automation to reduce their delivery time to complete more projects and deliver better user experiences. Ansible Automation Platform was at the center of that strategy.

The initial major automation project we supported this client with was their migration from a legacy, on-premise SAP system to SAP S/4 HANA® in the cloud. It's a big transformation project; it would usually take weeks of manual effort from many team members, plus it would come with added risk due to human error, which could lead to downtime. By using Ansible Automation Platform, deployment time went from 3 weeks to less than a day, which is a pretty dramatic time savings. Plus, it gave the organization the confidence that their systems would continue to work as normal during the migration so that the customer experience wasn't affected.

After the migration project, we started to work on applying automation to different parts of their in-store operations, like different software processes that keep their network services security-focused, their distribution centers running smoothly, and their point of sale system stable. These projects weren't just about reliability and time savings. They helped the company boost productivity, collaboration, and resilience. From there, the automation mindset really took hold. They held a hackathon to introduce more staff from different parts of the company to what automation can help achieve, and they established a **community of practice (CoP)** so that team members could share automation experiences and best practices.

This shift from a single automation champion working on isolated automation projects to using automation for organization-wide use cases to achieve time savings and boosting reliability didn't happen overnight. Making automation an essential element of IT operations requires a cultural transformation. It's not something you can just flip on and flip off. It takes practice, shared behaviors, and embracing a culture of change to transition to an automation-first mindset. But what you can automate is limitless.

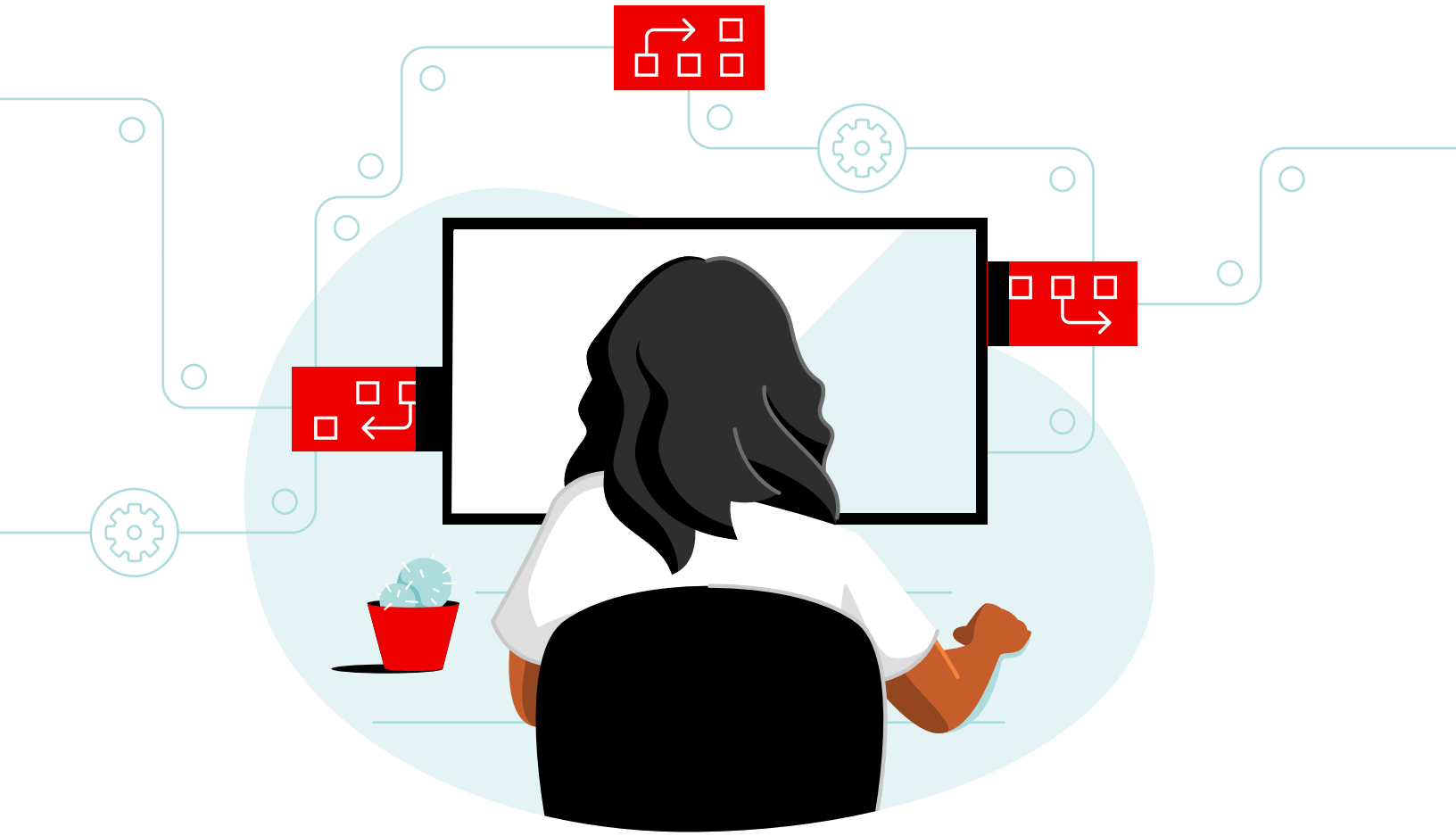
→ **Read the complete case study**

Jered Cortez

Jered is the Ansible Ecosystem Sales Leader for North America at Red Hat. He joined the company in 2021 after a successful tenure at IBM, where he oversaw technology sales growth in AI, business automation, and cybersecurity across multiple markets for the Americas. Jered holds certifications from Red Hat, Microsoft, and IBM, having completed IBM's prestigious Summit Program and Global Sales School. He earned his BA from Penn State University and an MBA from Elmhurst University.

Chapter 3

How to become an IT automation expert



The rise of the automation architect

Use these tips to advance your IT career and establish yourself as an automation architect.

By Joseph Tejal, Solution Architect, Red Hat

Most of us have seen it unfold before our very eyes. Coming out from the smoke and darkness of automation challenges, there's a new role emerging: the automation architect.

Automation was a hot topic starting in 2015, as open source projects like Ansible were on the rise and becoming more popular. There was a lot of excitement and interest, but there was also plenty of skepticism and much trepidation in operation centers. Fast forward to today, and automation is now at the core of technology strategies for most organizations. And the landscape continues to evolve and become more complex as we continue to focus on mission-critical operations and establish foundations for AI-enabled initiatives.

Now, let's go back in time and reflect on how automation architecture advanced to what it is today.

Starting an automation journey

In the beginning, operations teams and individuals found automation fun and easy. They learned how to automate some daily and routine tasks even if most didn't have a development background. Easy-to-learn tools like Ansible made it easy for most eager squads. Everyone from systems administrators to infrastructure and database engineers and even our service desk folks learned to automate.

Interest was growing, from the talks over coffee to hallway conversations to the practical demonstration to convince management. We were provisioning machines, restarting applications, performing patching and maintenance, and the list kept growing. People started to collaborate, form ideas, and talk about their cool stuff running on their space and how they can all work together.

But when they began to link small individual pieces of automation together, teams started to encounter problems. You probably know why.

Facing the challenges

Soon, things became more complex. There were more use cases to cover, and now automation routines and workflows needed to be interlinked. We began to see the challenges, and even worse, some individuals became victims of their own success as they were handed more significant requirements from management.

Here are some of the issues you may face with automation:

- Competing priorities and goals.
- Lack of standards, policies, and governance.
- Bottlenecks due to segmented work culture.
- Absence of some required skills.
- Implementation security concerns.
- Lack of consideration for reusability, scalability, and control.

Many operations teams turn these challenges into opportunities. When presented with some of these challenges, people initially feel confused and start to lose direction due to the lack of ownership and accountability. However, these challenges present an opportunity to take a leadership position to build successful automation practices, no matter how complicated the environments and tasks may get.

How the automation architect saves the day

The road to a holistic automation approach starts with an honest assessment of your **automation maturity**, followed by actionable steps across these core competencies:

These competencies include:

- Strategy
- Process
- Prioritization
- People
- Structure
- Operations

We need an automation architect to lead the organization on a journey so it knows which direction it should be going—and how to get there successfully. The automation architect's role is to address the challenges I mentioned above to help everyone get the full value of automation.

Some of the crucial tasks required to steer the organization effectively include the following:

Understand the current state of automation and set goals

This process gives the architect a good view of what existing tools and capabilities may be useful—and what else is required. It's imperative that you set achievable goals and identify priorities, which align to keep everyone on track and moving in the right direction, even if there are setbacks and detours. This effort also allows everyone to focus on the essential tasks ahead with the goals in mind.

Promote unity and collaboration

There will be continuing gaps, differing opinions, and conflicting working styles. The automation architect should bring everyone together by focusing and aligning the efforts to the business objectives and priorities. These efforts should translate into well-defined requirements that everyone can refer to and discuss openly. Use an open, collaborative approach and tools that encourage the team to work together. Incorporate activities to break down segmented groups and celebrate collective and inclusive success.

Improve automation assets and capabilities

Once you map out an automation inventory, centralize the resources in a core repository so everyone can start working collectively on these resources. Setting standards, proper review, and version control focusing on security and best practices provides a reliable method across teams. The team can now start working on capability gaps based on these tools and workflows using learning styles that are effective based on your team structure. There are now many free resources available to learn across various tools.

Shift everyone to advance the automation mindset and culture

Building automation advocates and champions is key to creating a holistic approach and bringing a shared understanding of how automation works and benefits teams. The automation architect should find creative ways to build trust, enthusiasm, and interest in automation. This approach can range from informal lunch meetings, success dashboards, and demonstrations, to a code-a-thon challenge that brings fun to the table.

Launch an automation community of practice

A CoP is a group of individuals within an organization who come together to share ideas, experiences, challenges, questions, and best practices around a technology. Establishing a CoP for automation can help teams working with automation to expand their automation skills, learn from each other's experiences, and identify new ways to collaborate across teams. Plus, it gives teams an opportunity to share and access automation content, so they can apply automation in a trusted, consistent, and repeatable manner. Learn more about setting up an automation CoP in the Speed automation adoption with a community of practice e-book.

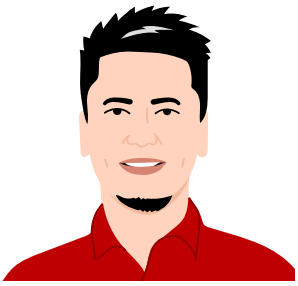
Scale automation solutions with proper governance and management

A good framework allows automation to scale and grow within teams without sacrificing security and best practices. The automation architect must establish governance adhering to the organization's policies, making people accountable through a solution with controls and robust auditing, while giving everyone the chance to succeed and innovate within properly managed and identified risks.

The established automation architect

It's inspiring to see the rise of new roles such as automation architects in this fast-paced era of evolving complex technologies. It's a great opportunity for sysadmins, subject matter experts, site reliability engineers (SREs), and engineers to make the most of their time across operations and automation journeys. It encourages them to level up and widen their horizons by going outside of their constrained fields of expertise through the power of automation. They become strategically positioned to understand and analyze the existing organizational challenges and pains. Then they can show how they address these issues by aligning solutions with business goals—collaborating with others and using the best approach and tools for successful end-to-end automation. To move forward in this role, you need to continuously grow and educate yourself so you can understand automation problems better and approach them with the right solutions.

→ To learn more, read The automation architect's handbook.



Joseph Tejal

Joseph is a Solution Architect at Red Hat with a passion for automation. As a dedicated advocate for streamlining processes, he focuses on enhancing service delivery by eliminating mundane tasks through the Ansible Automation Platform. With extensive experience in delivering automation projects for various clients, Joseph has contributed to the tech community by speaking at AnsibleFest and actively co-organizing Ansible and OpenShift meetups across New Zealand. His commitment to innovation and community collaboration inspire his mission to help businesses thrive through intelligent automation solutions.

8 skills you need to be successful in IT automation

Advance your automation skills with automation development, collaboration, source code management, and more.

By Chad Ferman, Senior Principal Product Manager, Ansible Business Unit, Red Hat

Looking back now, I was lucky ... I started working in an IT shop that, even before the year 2000, already had an automation team. This team had some sort of magic that proactively fixed issues before they happened—or at least addressed them when they happened without waking someone up in the middle of the night.

As a 19-year-old, I had no idea that this was not the norm. I didn't even really know what automation even meant. Then, it hit me: This is how I can make time for all of the other things in my backlog of work, and even more importantly, I don't have to do this manually ever again. The realization that anything you can do on a command line could easily be saved as code and run again systematically without human intervention completely changed my life and set me on the path I am on today.

The power of automation

In my last role, as the architect responsible for the **automation strategy** for an entire company, my mantra was: Automation is not just writing automation content. There is so much more to automation. Yes, scripts are the basis for automation, but they are not the only piece of it. To make something repeatable, you need information to tell you what state your systems are in and if they are behaving as they should be. This is where observability and monitoring come in. They let you make informed decisions about what needs to be done programmatically to accomplish your end goal. Once you have a feedback loop of information coming in and automation going out, you have a continuous cycle of improvement for your service delivery.

Why should I add these skills to be successful in my career?

Automation powers everything from application development to infrastructure deployment to business processes. The opportunities to add value are endless. You can work as a DevOps engineer, site reliability engineer, agile coach, product owner, integration engineer, AI/ML operations lead, or business process engineer, just to name a few. Understanding how

systems talk to each other to provide business value is a sought-after skill in many industries, and if you are a person who likes to create repeatable processes that work autonomously, then this is the job for you.

Doing more work with fewer people is something many organizations are dealing with, especially in economic downturns. By eliminating manual tasks, you make time for improvement. This approach builds trust with management through reliability and timely resolution of unexpected downtime. Successful patterns of automation can help everyone understand and unite around a common goal.

For example, I was given greater responsibility and opportunity by demonstrating successful patterns in automation. In a previous company, I was promoted from DevOps engineer to enterprise architect in 3 years by improving existing processes—we went from deploying servers in months to providing full stack application servers in 30 minutes. This was end-to-end deployment of business applications, configured, running, and providing business value.

Automation can be extremely rewarding because it's fun. It's a thrill to watch a 40-step CI/CD pipeline run and validate security and APIs, perform code analysis and linting, confirm that user interface elements are in the correct places, and perform regression tests that show the status of the pipeline in a dashboard as green/successful.

Knowing that what you do is valuable and repeatable by anyone else you work with is a great feeling. Also knowing that your changes will not affect others, and their changes will not break your work is a great feeling. Would you rather watch TV or act out a play yourself? You can relax and know that things will work as intended, or if they fail tests, let you know what happened so you can fix it later.

What skills do you need for automation?

Have you heard of minimal viable skills (MVS) for automation? These skills include but are not limited to writing automation content, collaboration, source code management, Kubernetes, security, testing, observability, monitoring, and network awareness.

Automation development

Having the ability to write automation content proficiently with your platform's built-in language (e.g., PowerShell for Windows or bash for Linux) is a great place to start. However, once you get into more complex automation, understanding a universal language like Python is desirable. I only call out Python as it has become more of a de facto standard for networking, server, storage, and AI/ML over the past 10 years. Entire automation frameworks have been written in it.

Recent innovation is making it easier for automation developers—as well as other roles, including those with less automation experience—to write automation content. Generative AI

(gen AI) technologies like **Red Hat Ansible Lightspeed** help automation teams learn, create, and maintain automation content more efficiently by turning natural-language prompts into code recommendations.

Collaboration

It takes multiple teams working together to unlock true end-to-end automation. Very few people know how the network, storage, firewall, proxy, and other elements really work, so there needs to be a commonality to tie all of these things together. This is why it is vital to have a company strategy for how and where you store your automation code and how you deploy it. Without a standard place to share not only code but architecture designs and APIs to communicate with each different part of the architecture, it is impossible to properly automate the delivery of infrastructure, applications, and services for our customers.

Source code management

Centralizing all code into a **Git** management tool such as GitHub, GitLab, Azure DevOps, or Bitbucket will make it much easier to collaborate with other teams and people in your group. Getting comfortable with putting in issues if you find a bug and documenting it well (meaning more than saying "it's broken") is a great way to get started if you are not ready to start committing code. Once you are comfortable with issues, begin making some pull requests and commit fixes to code or review pull requests that others have put in so you can help test their functionality. People really appreciate code reviews, as perfection is nearly impossible and you may see something they didn't consider.

APIs

Creating a centralized catalog of APIs and playbooks that everyone works from is crucial to automation success. This skill is more than simply having APIs in your applications. Having them available for anyone to use without calling you to ask how to interface with your service is the best way to automate service delivery. This way, when a developer needs a traditional infrastructure service, it becomes exactly like a cloud resource that they can request without having to put in a ticket or pick up the phone.

Containers and Kubernetes

Containers and Kubernetes have become the de facto way to deploy modern applications across a hybrid cloud. Having a solid understanding of how to build a container and then deploy, scale, monitor, and redeploy it is highly sought after by companies. This skill applies across many different parts of companies, from ML and application development to business intelligence and cybersecurity. With containers, you can ensure components that run on your local machine will work exactly the same on any platform at the application level. When you take the next step to deploy the container with Kubernetes, you can ensure via code that everything you need is in place for smooth deployments across all environments.

Network awareness

The way I always start the network awareness conversation is that the cloud does not solve physics, (i.e., the speed of light). It is critical to understand where data and users are located and where the data computation happens. First, understand the end-user's location compared to the data that they will be accessing. Make sure processing is in the same place as the user. Failing to do so is a common mistake that we see repeatedly that causes the application to perform poorly. Of course, the application gets blamed, not its poor placement.

Taking latency into consideration is another thing we take for granted, especially if you are in a country with good bandwidth. Make sure you conduct round-trip tests for latency to see what the end user's experience will be. For example, I had an application once that someone wanted to build in Texas even though all of the end users were in Singapore. After much deliberation, the application was replatformed in Singapore, and the user experience to get what they needed from the application went from minutes to seconds.

Testing

Testing is another skill that is dismissed as nice to have, but this is the thing that saves you from a single bad keystroke that takes down a production environment. Validating that the things you have set in motion perform as you expect them to is extremely important to being successful and making sure that you do not cause unintended consequences that will have you working through the night to resolve an issue. This should not be limited to back-end testing. There are great tools to test and validate user interface (UI) elements and APIs to make sure changes do not affect existing functionality or end-user experience. For example, Ansible development tools provide a suite of capabilities that can test Ansible modules, troubleshoot Ansible content, and identify and correct Ansible Playbook errors, amongst other functions.

With cybercrime on the rise, security testing is another role many companies are hiring for.

Security

Building security into your application is crucial in today's world of ransomware and bad actors taking over cloud deployments. Security integration should be part of the CI/CD pipeline that deploys the application. Within this pipeline, there are key things you need: static code analysis, artifact management and tracking, security-focused libraries, and code signing to make sure that when it is deployed, it is the same code or artifact that you think it is. CI/CD only covers the instantiation of the application. You also need to have security hardening on the platforms you are deploying to. In addition, you want something that

validates that you are not running a library that has a known vulnerability in it– and that can alert your team if the library needs to be patched.

With cybercrime on the rise, security testing is another role many companies are hiring for. Building security testing into your application supply chain is becoming something that many are doing throughout the lifecycle, from build to deploy and runtime validation that the code is executing only what it should be. Signed libraries and executables are becoming the norm, as is the validation of the sources of libraries and artifacts like containers. Using trusted signed libraries and containers and providing your organization with a custom library and artifact repository has become a standard for any security-focused organization.

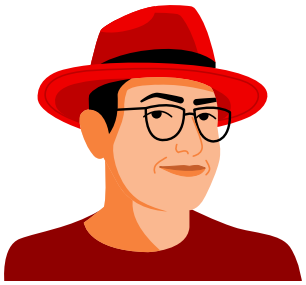
Observability and monitoring

Understanding the application state and how it achieved that state is another skill needed to automate tasks properly. Unless you know what is happening with your service, it is impossible to create proactive automation to fix issues or apply a consistent state that avoids the problem in the future. Most people stop at monitoring and then, if there is an issue, they use a root cause analysis (RCA) to discover what happened. Observability provides the tooling you need for an RCA, so you always have it and know what is happening well beyond the up or down status that monitoring usually leaves us with. The **DevOps monitoring guide** is a great resource to learn more.

Conclusion

I believe automation is among the most rewarding jobs a person can have in the modern IT world. It requires big-picture thinking and an understanding of how things work end to end. If you are a tinkerer and aren't satisfied with being told "that's just how it works," this is the job for you. Every time you take something that people do manually and make it into a repeatable process so they can focus on more valuable work, you save your company money (we all know a lot of the time this is what it really comes down to), as well as help people work on much more exciting projects. All of the skills listed above build upon each other to help you become a better automation expert. These skills are not acquired all at once, so take your time, enjoy the ride, and stop doing things manually.

*Adapted from **8 skills you need to be successful in IT automation**.*



Chad Ferman

Chad is a Senior Principal Product Manager for the Ansible Automation Platform at Red Hat. He has worked in enterprise IT in public and private sectors in retail and oil and gas for more than 25 years. His roles have covered everything from infrastructure operations to microservices application development and enterprise strategy. He came to Red Hat to help customers be successful in their enterprise software deployments and their cultural transformation into modern methodologies.

6 ways to increase your Linux sysadmin earning profile and potential

If you need a career or salary boost, here are some great tips to put you on the right track.

By Joseph Tejal, Solution Architect, Red Hat

Introduction

It's that time of year—you're sitting with your manager about to discuss your performance. Are you prepared to talk through this opportunity to increase your salary or be a candidate for promotion?

The best case scenario is that you don't have to do much talking. Your achievements and the value you've added to your organization speaks for itself, and your manager thanks you for making their life easier, justifying your advancement. This is just 1 way you increase your earning potential as sysadmins—by growing within your organization.

When other opportunities come from outside your workplace, are you prepared with your success stories to convince potential employers and get them to buy your pitch during the interview? The ideal case is that you effortlessly share your initiatives and successes, they listen with interest and want to hear more, and then end up hiring you with a good offer. Better yet, your profile and brand in the local community are so outstanding that different companies battle it out to win you.

These are some of the possible ways to increase your earning potential as a sysadmin. The big question is: how do you get there and prepare yourself for these conversations and opportunities?

Here, I share some of the tips, advice, and ideas based on my experiences and those of others I've spoken to.

Make yourself dispensable

I know this sounds counterintuitive, but I've learned that to move forward, you need to leave your existing tasks behind and focus on adding value.

Add value to your team and organization

Improve, automate, and document your day-to-day tasks so that anyone can do it. Better yet, so that not a single person needs to do it. Target to fix your main pain points to make

everyone confident in your space. This way, you have time to participate and gain everyone's trust for you to become involved in more valuable initiatives.

Don't stick to the status quo. Challenge yourself and the norms, especially if they're inefficient and outdated. Your stakeholders will be impressed with the value and improvements you're making.

Take control of your career and goals

Managing your career and goals allows you to conquer your own limits and your organization's. You sometimes hear that you don't get the support you need, but by taking initiative, you can learn and move forward through the resources available to you. There are numerous open source projects that you can experiment with even before you request technical training.

Do the research and read proof-of-concept studies on technologies such as Red Hat Ansible Automation Platform, hybrid cloud, Kubernetes, and Red Hat OpenShift® using the free trials and workshops available online. These resources help you become a good candidate for future learning and development investment by your company, helping you qualify for formal training and certification exams.

Managing your career and goals allows you to conquer your own limits and your organization's.

You may want to find a learning buddy, technical coach, or mentor, as it may be easier to commit when you're accountable to someone else. It's not just technical development—you might also want to grow vertically into leadership and manage other sysadmins.

Work on your soft skills

In this era, having technical skills is not the only way to increase your earning potential. Emotional intelligence, attitude, the ability to work with others, and communication skills are some of the things that can give you a unique advantage and distinction.

Begin by looking for a peer or coach that you respect in your organization. You can also join speaking clubs such as Toastmasters to enhance your interpersonal and communication skills in the meeting room and on a speaking stage. Writing is also a way of getting noticed. There are many venues, such as opensource.com and medium.com, where you can contribute and learn from others.

Don't be overwhelmed or afraid to get out of your comfort zone. I know many people who surprised themselves with their strengths away from their keyboard, terminals, and technical expertise.

Collaborate with others

Gone are the days when some sysadmins know the top secrets and gain advantages, and everyone else remains left out of critical knowledge.

We live in a world where innovation springs out of collaboration. Be sure that you work with other teams to expand your knowledge and contribute to theirs. You might even take the initiative to do rotations on teams that interest you. Try to organize small projects that improve some processes within teams or explore updated tools and technology that may be better than what you currently have. This collaboration helps to break down divisions, start a cultural change, and spark interests, allowing everyone to succeed.

Ensure everyone's work and progress are documented publicly for easy access to those who were encouraged to participate.

Connect with others and increase your profile

Career connections and references give you an advantage. Great feedback from key people you've worked with gives your manager an idea of how you perform and exceed expectations.

Being part of communities also allows you to connect with different people who may be a potential employer. **Attend or organize events**, such as lunch and learns or meetups within your company or local community. Getting yourself out there helps others and helps you get noticed.

You might even start with small groups or lunch sessions within your team or department to discuss your successes and exciting projects. Once comfortable, you might want to submit papers or proposals to tech conferences and engagements about the cool things you are working on—this further raises your profile.

Have fun at work

Find the things that fuel you. Work can be daunting at times, but it's how you react to difficult times that define you. If you're enjoying your career or significant aspects of it, you're excited to be in these situations where you can help and add value.

Make it fun and look for opportunities to display your skills and strengths, and include some opportunities to develop your weaker points. Work isn't boring when you expand your horizons and discover newfound enthusiasm.

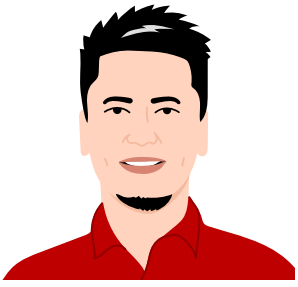
There are many ways to make it exciting by connecting with others, being creative in your space, and reaching out for opportunities outside of your normal routine. Make sure to take recharge days outside of work to renew your energy.

The future is bright through collaboration

Your ability to earn and sustain yourself is a key motivation for why you work. Increasing this earning potential alongside career growth, personal fulfillment, and enjoyment gives you the impetus to go further. Make sure you have a success narrative ready to share with others. But keep in mind, it's not an overnight process. It's all of the hard work, investment, commitment, and enthusiasm that you put in daily with your goals in mind.

Take every chance you can to build your brand and story, so when the opportunity comes, your narrative speaks for itself and helps guarantee your success.

Adapted from *6 ways to increase your Linux sysadmin earning profile and potential*.



Joseph Tejal

Joseph is a Solution Architect at Red Hat with a passion for automation. As a dedicated advocate for streamlining processes, he focuses on enhancing service delivery by eliminating mundane tasks through the Ansible Automation Platform. With extensive experience in delivering automation projects for various clients, Joseph has contributed to the tech community by speaking at AnsibleFest and actively co-organizing Ansible and OpenShift meetups across New Zealand. His commitment to innovation and community collaboration inspire his mission to help businesses thrive through intelligent automation solutions.

6 ways to pivot your team to an automation-first mentality

DevSecOps can provide a competitive edge for your organization. Use these 6 strategies to get started.

By Piyush Patel, Senior Managing Architect, and James Brokenbek, Senior Portfolio Manager

Automation is not a new concept. Most network admins have long been doing some level of automation, in some way, shape, or form—things that make their job easier. But this kind of isolated automation doesn't deliver long-term value. In order to expand automation across an organization—and therefore realize greater value—an automation-first mentality is required.

What is an automation-first mentality? It's about actively looking to automation as your go-to, primary solution for improving manual processes and workflows in ways that reduces toil and drudgery, and boosts efficiency and productivity.

But getting to this mindset isn't easy. Whether it's resistance to change, an inability to think beyond existing processes, or simply being stuck in reactive mode, there are real obstacles to adopting an automation-first mentality and getting others onboard. In this section, we're going to dive into 6 strategies that can help you get there.

Find an automation champion

Successfully shifting mindsets requires someone to lead the charge and inspire others. This person is your automation champion: someone who is excited about the potential of automation and wants to share that excitement with others. They have a vision for how automation can deliver meaningful benefits for your organization, and they're passionate about bringing others into that vision—through simple hallway conversation, hosting events, or sharing their own results with small use cases.

Look beyond technical competence. A good automation champion candidate should have some level of know-how, but more importantly, they have some level of influence in the organization. They may not be in a leadership role, but they are well-liked, respected, and trusted; others want to hear what they have to say.

Focus on small, early wins

Resist the temptation to tackle the biggest, most important automation use cases. Successfully automating smaller, single task projects can help you create a proof point and combat the fear, uncertainty, and doubt that folks who aren't necessarily on your side may be spreading. If the ROI is small, but the investment was small, too, that's a healthy ratio and compelling result that can help generate buy-in from other teams.

These small, early wins are like Lego blocks that help you demonstrate value to other teams and establish a foundation for bigger, more complex automation projects.

Small, early wins are like Lego blocks that help you demonstrate value to other teams and establish a foundation for bigger, more complex automation projects.

Empower teams with training and mentorship

Organizations should look at what's going to help them be successful in the long term. Providing automation training and certification opportunities to your teams can address any skill gaps and help them feel more confident in deploying automation, which is critical to establishing an automation-first mentality.

Training and certification can be complemented by strategic consulting engagements. Consultants can work alongside you to establish an automation strategy that meets your goals and provide mentorship to you and your teams, so that you can become the masters of your own automation destiny. These engagements are like the training wheels that prepare and empower you to independently move forward with confidence.

Create an internal automation community

An automation-first mentality isn't effective if it's only held by a single person. There needs to be community adoption. But communities, like gardens, only flourish if they're tended to. And this is where an automation CoP can help.

A CoP brings people together—ideally from various teams, and with varying levels of automation experience—to discuss automation. Use cases they've recently tried, challenges, and better practices. It's a forum for sharing, collaborating, and then leaving with new learnings. A CoP can build camaraderie, rapport, and ultimately that translates to trust, which can help individuals and teams be more effective.

The most productive CoPs have a few formalized roles. An executive sponsor, who can help build a bridge between the CoP and corporate governance. A community manager, who is well-versed in automation and who can help organize logistics, guide discussions, and help generate consensus. And then someone whose role it is to help onboard new members from different teams.

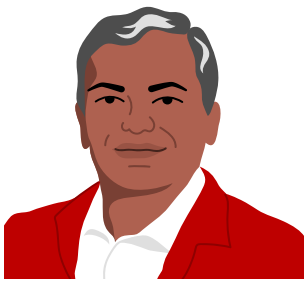
Establish a single source of truth

An automation-first mentality requires a readiness to automate new tasks and processes and implement repeatable automation practices that can be shared across the organization. Establishing a single source of truth, using code as your documentation, ensures everyone is working from the same, trusted information, which makes collaboration stronger and reduces the likelihood of errors. From the get-go, you need to think about where information resides, whether there is an authoritative listing of that information, and how that information flows to other parties that need it.

Think of an org chart that lays out a company's employee hierarchy and reporting relationships. Depending on when they started at the company, employees might have different versions of an org chart, which can cause issues like emails being sent to the wrong recipient. Ultimately, an HR specialist is maintaining an up-to-date version of the org chart—that's the source of truth. And if everyone has access to that same source of truth, then a lot of mistakes can be avoided.

Plan for automation at the beginning

You'll know your organization has successfully adopted an automation-first mentality when your teams consistently plan for automation at the outset of a project, rather than after a process has been implemented. To plan for automation at the beginning requires analyzing the key tasks and actions within a project, and then breaking them down into elements to be automated. By embedding automation into a project at the very start, you can move faster, reduce bottlenecks, and be more responsive.



Piyush Patel

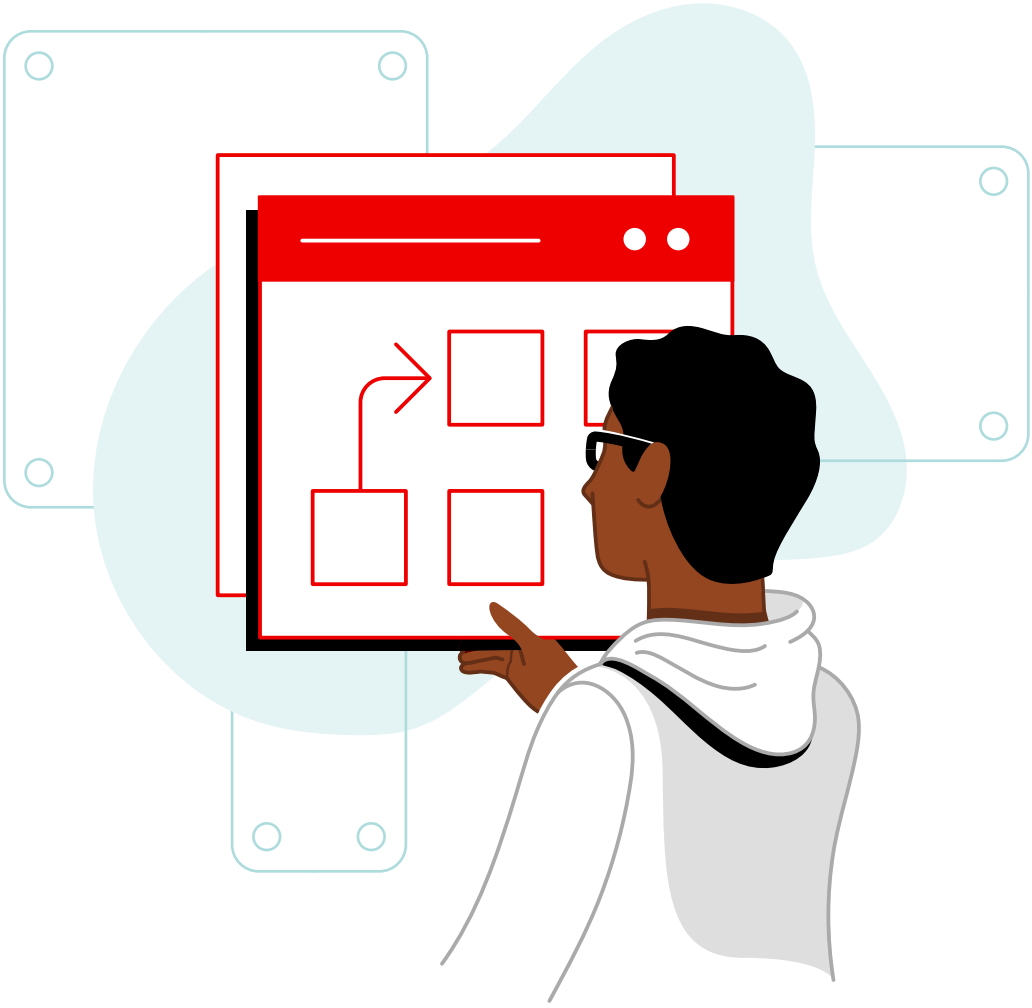
Piyush is a Senior Managing Architect in Red Hat's Ansible Automation Platform Practice. He joined Red Hat in 2017, working with the global telecommunications team before moving to automation. He is passionate about automation and has a strong background in all aspects of datacenter, virtualization, networking, and cloud. His motto is: "If you are going to perform a task more than twice, then spend time to automate it."

James Brokenbek

James is a Senior Portfolio Manager, focusing on Red Hat Services engagements that support Red Hat Ansible Automation Platform. He joined Red Hat in 2014. With a background in complex technology systems and a passion for empowering shared success, James helps customers take thoughtful approaches to automation.

Conclusion

How to get started with IT automation



Embracing IT automation: Why it's good for your career—and how you can get started

Red Hat can make the path to automation more efficient with comprehensive training options that translate into immediate business benefits.

By James Mighion, Principal Site Reliability Engineer, Ansible Cloud Services, and Steven Bonneville, Principal Portfolio Architect, Red Hat Training

In 2021, Lietuvos Geležinkeliai (LTG), Lithuania's state-owned national railway operator, transported 4.13 million passengers and 51.1 million tons of freight. But behind the scenes, it lacked a common IT architecture, which made management difficult. LTG set out to standardize its IT environment and adopted Red Hat Enterprise Linux as its foundation, and Ansible Automation Platform to simplify complex hybrid deployments and automate repetitive management tasks. To set teams up for success, LTG also recognized that upskilling was needed to support its staff to effectively manage these new platforms. They implemented Red Hat Training to help staff build their skill sets, learn best practices, and apply their knowledge.[6]

The benefits of Red Hat Training and Certification apply to individuals, not just organizations. Many IT professionals seek automation training on their own and find that it enhances their value to their company and their personal marketability. Take, for example, Christian Sandrini, an IT professional who was named the 2021 Red Hat Certified Professional of the Year. He has earned nearly a dozen Red Hat certifications, including the Red Hat Certified Specialist in Ansible Best Practices and the Red Hat Certified Specialist in Ansible Automation Platform.

Sandrini participates heavily in Red Hat Training as a Red Hat Learning Subscription subscriber. "The Red Hat Learning Subscription has helped me because it suits my learning style," Sandrini said. "I enjoy having the self-paced learning platform where I can go back to rewatch certain videos or do the labs, which are very hands-on. It helped me get into the technology very fast, so even if I didn't know about a certain product, I could sign up for a class and learn very quickly what the product is all about. Another thing I really enjoy is the early access section of Red Hat Learning Subscription, because it gives you a sneak peek into the technologies that are coming next."

6 Red Hat case study. "LTG uses SAP-certified Red Hat platforms to optimize IT." 13 Jan. 2023.

Sandrini's initial step toward implementing his knowledge was introducing his organization to Ansible Automation Platform. His goal was to automate functions and modernize how the team provisions and configures servers with an IaC approach. After receiving overwhelmingly positive reviews of his work from the team, he deployed Ansible Automation Platform. That was really where the change happened," Sandrini said. "It allowed us to have role-based access control, and this is when other teams started to become interested. It shows that Ansible cannot only be used for Linux, but for other components around it as well."

As the demand for professionals skilled in IT automation continues to grow, so does the importance of staying abreast of new strategies and technologies.

Sandrini has become an invaluable asset to his own team, implementing and maintaining IT automation to streamline processes and cut costs using the knowledge obtained through Red Hat Training and Certification. He advocates making training available across teams to acquire and improve skills, as well as to distribute knowledge across team members.

As the demand for professionals skilled in IT automation continues to grow, so does the importance of staying abreast of new strategies and technologies. In the ever-evolving field of IT, training and professional development is key to remaining competitive on an individual and organizational level. Red Hat Training and Certification is constantly updating course content to keep pace with the industry, ensuring that trained and certified professionals who want to enhance their careers in IT automation have the tools they need for success.

For more than 20 years, Red Hat has provided training and certification options for technology professionals to keep up with current and emerging trends. Red Hat Training courses have options to suit various needs and learning styles, from traditional, classroom-based training to onsite training to diverse virtual learning options. With an average of 20 years in IT and 11 years using Red Hat products, Red Hat certified instructors strive to create an immediate effect on business goals with their expertise.

Red Hat Learning Subscription provides on-demand access to the complete catalog of self-paced courses, videos, and labs. All of this content is consistently updated to keep pace with the industry. In fact, nearly a quarter of courses available in the Red Hat Learning Subscription catalog have been added or updated within the past year. Some Red Hat Learning Subscription tiers also include the cost of sitting for a Red Hat certification exam, simplifying the journey from obtaining knowledge to proving it.

While Red Hat Training helps students keep pace with the latest in automation and digital transformation, Red Hat Certification validates IT professionals as skilled and prepared to take on the most ambitious projects in the face of evolving industry challenges. Red Hat

certified professionals help their organizations achieve optimal efficiency and cost savings through expertise in automation. Plus, organizations are drawn to recruit and retain certified professionals because of the reduced onboarding time and increased cost savings.

Red Hat currently offers 6 certifications with a focus on automation:

- **Red Hat Enterprise Linux Automation with Ansible (RH294)** and the **Red Hat Certified Engineer (RHCE) exam (EX294)** lay the groundwork for automating workflows, employing DevOps practices, and using Ansible Automation Platform for more efficient development.

- **Developing Advanced Automation with Red Hat Ansible Automation Platform (DO374)** and the **Red Hat Certified Specialist in Developing Automation with Ansible Automation Platform exam (EX374)** exam build on these Linux automation skills with more advanced Ansible techniques.

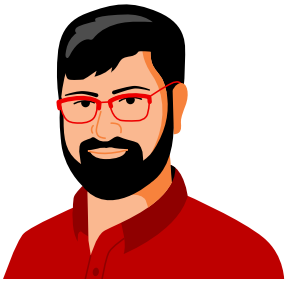
- **Red Hat Services Management and Automation (RH358)** and **Red Hat Certified Specialist in Services Management and Automation exam (EX358)** show IT professionals with some experience managing Linux systems how to apply automation to the management and deployment of network services included with Red Hat Enterprise Linux.

- For network administrators, **Network Automation with Red Hat Ansible Automation Platform (DO457)** and the **Red Hat Certified Specialist in Ansible Network Automation exam (EX457)** provide an entry point for automation of routers, switches, and other networking hardware from multiple device vendors.

- For Windows server system administrators, **Microsoft Windows Automation with Red Hat Ansible Automation Platform (DO417)** and the **Red Hat Certified Specialist in Microsoft Windows Automation with Ansible exam (EX417)** provide an entry point to learn Ansible automation in a Windows-oriented context.

- **Managing Enterprise Automation with Red Hat Ansible Automation Platform (DO467)** and the **Red Hat Certified Specialist in Managing Automation with Ansible Automation Platform exam (EX467)** develop and prove the skills needed to use and extend existing Ansible infrastructure across business units in large enterprise environments.

What else can help get you up and running faster with automation, in addition to training? AI. Using gen AI capabilities, Ansible Lightspeed simplifies the process of creating and running automation tasks. Its ability to turn natural language commands into Ansible Playbooks or tasks helps lessen the learning curve and makes it easier for users of all experience levels to start contributing.



Steven Bonneville

Steven is a Principal Portfolio Architect at Red Hat in its training organization. With more than 20 years working at Red Hat, Steven provides deep expertise with Red Hat technologies to design courses for Red Hat's training curriculum, mentor curriculum designers and developers, and recommend directions for future training development across all products in the Red Hat Training portfolio. For many years, he was responsible specifically for the Red Hat Enterprise Linux system administration and Ansible automation curricula, including the RHCE® training track, and advanced courses on system administration, virtualization, and storage. He wrote the 1st version of many of those courses.

James Mighion

James is a Principal Site Reliability Engineer in the Ansible Cloud Services organization. He's been with Red Hat since 2011 and has had many different roles along the way. James is passionate about automation and contributing to open source projects.

Get started

Start your trial of Red Hat Ansible Automation Platform

Get a single, self-supported 60-day subscription for Red Hat Ansible Automation Platform for Red Hat Enterprise Linux.

→ Try Ansible Automation Platform for 60 days

Learn the basics of Red Hat Ansible Automation Platform

Try our series of on-demand online videos. Ansible Essentials: Simplicity in Automation Technical Overview (DO007) introduces you to Ansible Automation Platform, including configuration management, provisioning, deploying, and managing compute infrastructure across cloud, virtual, and physical environments.

→ Learn about Ansible Automation Platform with free training

Lead with automation and connect your teams

Digital leaders aren't just automating their existing workflows—they're learning to create shared value with automation. Your organization may be automating some aspects of IT, but are you in an advanced or beginning stage? Take the online assessment to find out your current automation maturity, identify next steps, and get resources to support your progress.

→ Take the assessment