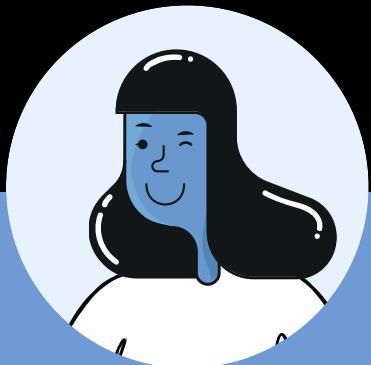




# What is Runbook Automation?



**Short answer:** Runbook Automation is the ability to take action.

**Longer answer:** Runbook automation provides self-service access to the operations capabilities you need to take action and get your job done.

**Even longer answer:** Keep reading.

---

## **What will Runbook Automation do for your operations?**

Less waiting – Replace "open a ticket and wait" with "here's the button to do it yourself."

Fewer interruptions – Cut down on the repetitive requests that consume your team's time and delay other work

Shorter incidents – Enable those closest to the problem to take action quickly and effectively.

Fewer escalations – Avoid the costly and disruptive escalations that disrupt your already overworked subject matter experts.

## **Life before Runbook Automation**

Moving the bits around is the easiest part of operations. You already have the tools, scripts, and manual commands that will copy artifacts, manipulate files, and call APIs.

The problem is that the knowledge needed to invoke and leverage those tools, scripts, and manual commands lives in the heads of only a few people in your organization.

That leaves everyone else with only a few unsatisfactory options:

- 1** **Brave the wiki** – Search for the correct docs and try to decipher what the writer intended (likely wondering the whole time if it is up-to-date and accurate).
- 2** **Dive into ad-hoc script/tool usage** – Look in previously agreed-upon locations for shared scripts and hope your knowledge of the correct usage and environment details is current.
- 3** **Escalate!** – The most likely option. Send disruptive interruptions deep into your organization and wait for a response.



It's a lack of up-to-date knowledge — or sufficient access privileges — that block others from participating directly in operations activity.

Consequently, everything (provisioning, incident management, diagnostics, maintenance, reporting, and more) falls to a few already overworked and bottlenecked subject matter experts.

This inability to allow more people to participate in operations leads to expensive and painful problems:

- Bottlenecks form around your subject matter experts.
- Incidents are longer than they need to be because only a limited number of people can take action.
- Escalations are rampant, causing more disruption and interruption, which in turn crowds out planned business improvement work.

## Runbook Automation To the Rescue



Runbook automation is essential to your operations because:

- Operations is more than executing a single command – You are routinely dealing with multi-step procedures that span multiple command line or graphical interfaces.
- Knowledge transfer is difficult and expensive – You have to convey what to do, the correct sequence, and how to evaluate the output at each step.
- Everything is changing faster – Under the pressure of Digital Transformation, DevOps, and Cloud-Native architectures, the pace of change (and complexity) has increased exponentially and will continue to do so. Timely and accurate knowledge transfer via meetings or written text is increasingly improbable.

Need to perform a restart or other action during an incident? Use an automated Runbook and ensure the most up-to-date procedures are executed.

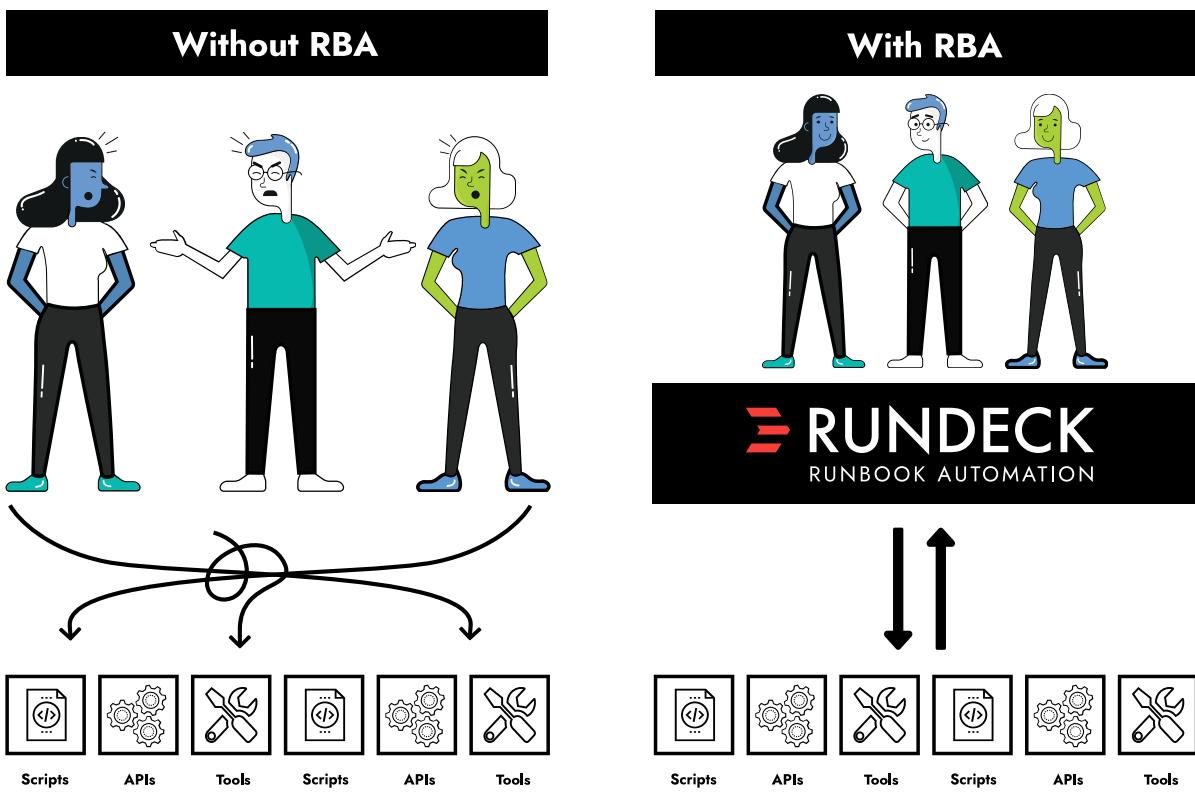
Want to refresh an environment or have new resources provisioned? Don't fill out a ticket; serve yourself using an automated Runbook.

Keep getting interrupted to check the health or performance of a production service? Create an automated Runbook so others can check the health or performance themselves.

Runbook Automation enables you to easily translate expert operations knowledge into automated procedures that anyone in your organization can execute on-demand (assuming they have the access privileges).

# RBA Leverages Your Existing Skills and Investments

To be clear, the role of Runbook Automation is not to replace your existing tools, scripts, API calls, or manual commands.



The role of Runbook Automation is to automate the workflows that span and invoke your existing automation and manual commands.

Runbook Automation quickly becomes the human-to-tool interface for your operations procedures.

# **ROI of Runbook Automation**

Calculating the ROI of Runbook Automation is dependent on the activity. There are two general categories: Incident Management and Provisioning.

## **Incident Management**

**Shorter Incidents** – Incidents cause lost revenue, opportunity cost, and reputational damage. By responding quicker and enabling a broader set of colleagues to respond, incidents are resolved quicker, and potential damages are decreased.

**Fewer Escalations** – Your people are your most expensive assets. By enabling people closer to the issue to diagnose and resolve the issue, you are avoiding the highly disruptive escalation chains that interrupt other work.

**Fewer Incident Response Hours** – Where does your organization spend its time? With shorter incidents and fewer escalations, your teams are spending less total hours responding to incidents and more time on project work or initiatives that move the company forward.

## **Provisioning**

- **Less waiting** – Stay out of people's way. Your team is spending less time filling out tickets and waiting for others to do something.
- **Fewer interruptions** – Protect the limited capacity of subject matter experts. Avoiding interruptions from repetitive requests gives your subject matter experts more time to work on the projects that move the needle for your business.

## **Critical Capabilities of a Runbook Automation Solution**

At the technical core, Runbook Automation is an interface to a workflow. However, there are a few essential capabilities needed for a successful enterprise solution.

**Automation Harness** – A universal hub that connects any scripts, tools, or APIs into a workflow. Works with any scripting language or tool and allows you to leverage your organization's existing skills and investments. If one team loves Ansible, drop in their playbooks. If another team is all PowerShell, drop in those scripts. The Automation Harness lets you plug in what you've already got (including manual system commands), and then use simple configuration to define the desired workflow.

**Guardrails** – Providing users with safe and controlled access to smart choices. These "Guardrail" features generally fall into two categories: access control and usability. Access control features constrain what users are allowed to do and provide a clear audit trail.

Usability features are focused on guiding users and reducing training requirements. Usability feature examples include dynamic options, user input validation, output formatting/processing, error handling, and conditional notifications.

**Dynamic Infrastructure Map** – Today's infrastructure and software components are continuously in motion. Whether you are responding to an incident or completing a provisioning task, you need to know the location and the state of the things you care about. A Dynamic Infrastructure Map keeps track of the details by integrating with other "sources of truth" in your environment (CMDBs, config management, cloud/VM managers, monitoring tools, and more). Now, the targeting of your automation and variables in your automation automatically stays up-to-date.

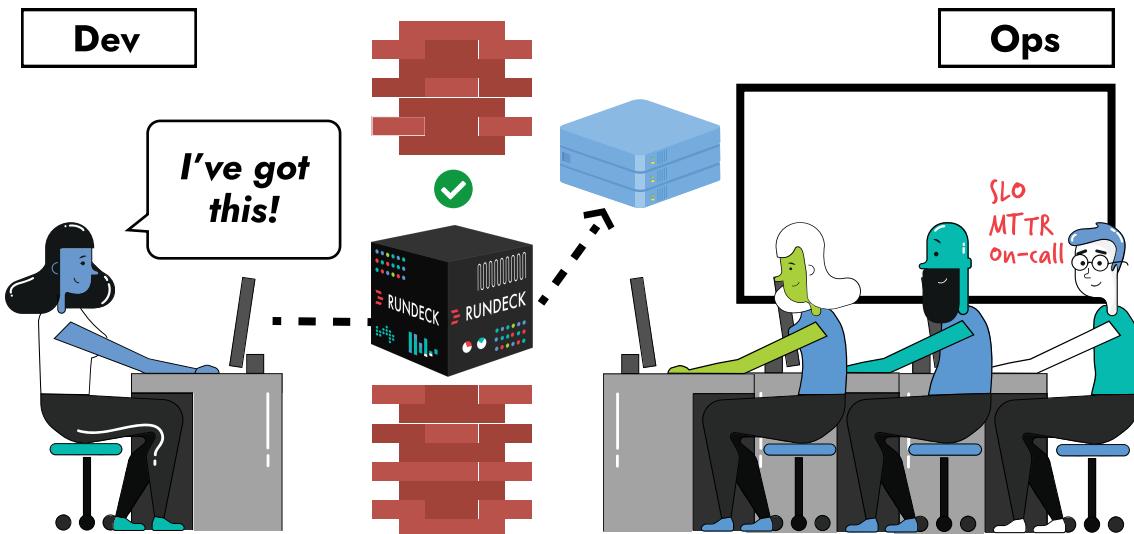
## **Runbook Automation for DevOps**

DevOps inspired ways of working are encouraging the delegation of operations work to those who are outside of the traditional boundaries of "Operations." For example, allowing Developers to deploy, investigate, and fix their applications in production under a "you build it, you run it" model.

Runbook Automation helps DevOps ways of working in several ways, including:

- **Enable Developers with safe, self-service access to do the "run it" part of "you build it, you run it."**

- Make it easy to hand-off operations procedures in a fast-moving or continuous delivery environment.
- By providing a secure, auditable platform through which all human-to-tool interaction takes place, making security and compliance comfortable with significantly expanding the number of people doing operations work in production.



## Runbook Automation for SRE

SRE (Site Reliability Engineering) is a significant change in how Operations work gets done. SRE emphasizes using software engineering practices to managing and improving the reliability, scalability, and performance of business-critical systems.

Runbook Automation helps SRE practices in several ways, including:

- Turning what would previously have been written documentation into executable code managed through a software development lifecycle.
- Self-service that enables operations activity to be distributed throughout an organization, reducing "toil" (a key SRE mechanism for regulating workload).
- Building and collaborating on automated checklists that improve the speed of diagnosing and resolving incidents.

## **Runbook Automation for Legacy Environments**

Life in enterprise Operations will always be a mixture of "the old" and "the new." Responding to incidents or doing a provisioning activity will – more often than not – require you to work across multiple generations of technology.

Runbook Automation helps operating in legacy environments in several ways, including:

- Capture standard operating procedures for all services, ensuring quick and reliable access for anyone responding to incidents or who need something provisioned.
- Go faster, but maintain ITSM standards, by replacing the need to open tickets for standard changes with self-service automation (that can still keep records in the ticket systems if needed).

- Ensure that the operations actions executed were the same as those previously agreed to during change advisory/review (with audit logs that allow you to review what ran, who ran it, and the output/results).

# **READY FOR SHORTER INCIDENTS AND FEWER ESCALATIONS?**

*Let's talk.*  
***hello@rundeck.com***

