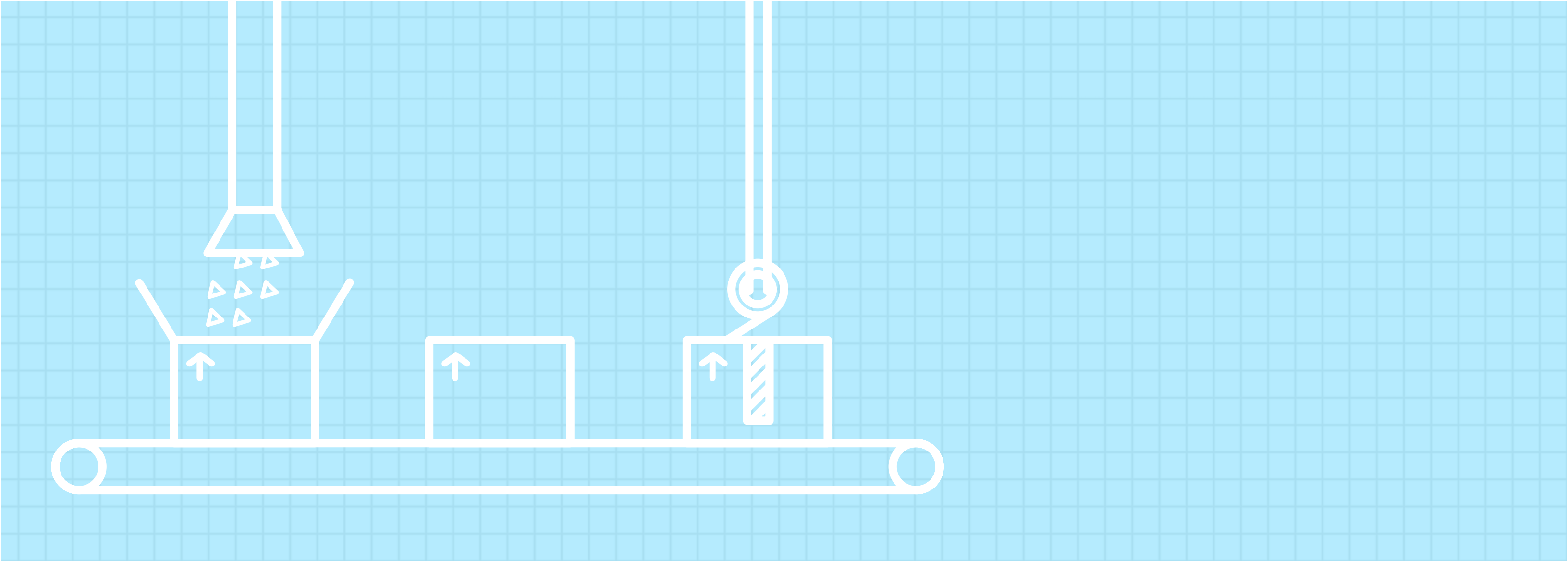
XebiaLabs

Benefits of Agentless Remote Automation



Engineering Group, XebiaLabs

www.xebialabs.com

Introduction

Deployit is the only Application Release Automation solution that does not require one of its components to be installed and running on every target system. Our unique agentless remoting capability is built on innovative, industry-acclaimed software engineering. Deployit's agentless approach is the result of careful consideration and decades of remote administration expertise.

In this technical research note, we explain the drivers behind this decision and outline some of the resulting benefits.

1. No components running under privileged accounts

Many agents are written on the basis that the most painless way to ensure they are able to carry out their task is to run as a privileged account on the target system, typically root or Administrator.

This is especially the case if the agents support impersonation of "normal" users using the equivalent of *sudo* or *runas*, while agents that do not support impersonation may require you to run multiple agents on a single target system simply to be able to execute commands using different credentials.

Any process running on a production system as a privileged account represents a potential security risk and should be extensively tested and audited for security issues. By not requiring its own privileged process on target systems, Deployit avoids this issue entirely.

2. No network chatter

One potential advantage often touted for polling "pull-based" agents is that they do not require any inbound connections to the target machines. This is certainly true; it is usually more of theoretical than practical interest, however, since most networks feature at least one administrative machine or "bastion host" that is externally accessible for general maintenance tasks.

Deployit's agentless architecture can tunnel through the bastion host – or multiple levels of bastion hosts, if necessary – to the target systems. This means that no new incoming or outgoing connections to and from the target hosts need to be permitted.

Moreover, polling agents cause a significant amount of network "chatter": think thousands of target hosts repeatedly querying the automation server. Reducing the polling interval can lower the "chatter rate", but reduces efficiency by increasing the average deployment time.

3. No process or dependency overhead

Writing a powerful yet efficient agent requires substantial engineering effort. As a result, many commercial agents are conspicuously heavyweight, even when idle, tying up memory and other system resources that should be powering your business applications.

But the intrusion footprint of agents extends beyond runtime use of resources: most agents require additional frameworks or components to be deployed on the target systems. This leads to decidedly suboptimal systems management scenarios, such as having to install a Java runtime just to deploy a .NET application, or installing the .NET framework to manage a Java EE app server. Unsurprisingly, it also represents additional security and maintenance challenges.

4. Proven secure channels

In SSH and WinRM, Unix and now also Windows systems provide an out-of-the-box, recommended solution dedicated to remote automation. These programs have been specifically designed for remoting by specialist teams whose in-depth knowledge of the underlying operating systems and their security architecture makes these teams by far the best qualified for this undertaking.

Moreover, SSH and WinRM, the default protocols used by Deployit for remote automation, have undergone years and even decades of improvement in both security and functionality. These programs are better documented, better understood, more thoroughly vetted and far more battle-hardened than any commercial agent relying on often unknown, untested and unverified proprietary code.

SSH and WinRM have been and continue to be subject to intensive scrutiny and peer-review, and have been cleared for use in highly secured environments. Any remote automation component with the same ability to execute arbitrary commands on a production system should be investigated with similar thoroughness before use.

5. No maintenance overhead

Even if there were no concerns at all surrounding the security, runtime load or system requirements of agents, the maintenance overhead of the associated ongoing housekeeping alone can be enough to render agents infeasible.

Installing, troubleshooting, restarting and updating agents, associated keys and certificates etc. across a variety of operating systems on potentially thousands of target machines is a significant administrative effort. No virtual applications, default VMs or AMIs are usable out-of-the-box.

Agent-based automation creates a clumsy meta-deployment problem, in which you need to deploy an application just so that you can deploy your applications. Ironically, this "bootstrap" deployment will generally use the standard remote automation interfaces (SSH or WinRM) that are all Deployit requires to deploy *all* your applications.

6. 100% transparent and reproducible

One of the key criteria when introducing automation for mission-critical activities such as deployments to production is that the commands executed by the automation solution can be easily and accurately executed manually.

This is not just important for disaster recovery and troubleshooting, it is essential to ensuring your technical experts have a clear and correct understanding of what the automation will do, and can reproduce and simulate its behaviour at any time.

Better still, by being able to replicate the exact procedures and commands already in use in most organizations, Deployit is able to transparently introduce automation without any change whatsoever to the existing administrative process.

About XebiaLabs

With customers such as 3M, John Deere, Duke Energy, Xerox, and Société Générale, XebiaLabs is a leading provider of delivery automation focused on helping organizations deliver business value faster and more efficiently. The XL platform combines Build, Deployment, Provisioning and Release Coordination to help DevOps and Release teams deliver higher quality software faster. Headquartered in the U.S., XebiaLabs has a world-wide network of sales offices and partners.

For more information, please visit **www.xebialabs.com**