✅ CI/CD Comparison: Using Managed Providers vs Self-Hosting

Posted June 8, 2018   👁 4.8k   CI/CD   CONCEPTUAL

By: Justin Ellingwood

Introduction

Continuous integration, delivery, and deployment are a collection of practices designed to help teams produce more reliable software quickly. While CI/CD is primarily a set of principles and methods, tooling plays a major role in making the ideals of the philosophy a practical option.

CI/CD systems help teams integrate, or incorporate new changes into software, more frequently by automatically building or packaging each commit to shared repositories. They run test suites against each new build and automatically promote it to more selective environments when the tests pass. Versions of the build that traverse the entire pipeline can be automatically deployed to live production systems or queued for manual deployment. Because of their broad responsibilities, CI/CD systems usually touch large portions of internal environments and infrastructure.

Choosing the right CI/CD system can be difficult, but generally the options fall into one of two categories: managed CI/CD services accessed through a provider, and self-hosted services that you can set up and manage yourself. In this guide, we'll discuss how these two main groups of software differ across different categories and mention some mixed approaches that attempt to offer a good middle ground. The best software for the job will be highly dependent on the needs of your organization and the capabilities and priorities of your teams.

Managing Infrastructure

One of the major differences between using a managed CI/CD service and running your own self-hosted services is infrastructure management. CI/CD systems require resources to run and, as with any infrastructure, this means that someone has to maintain the health of the services and the underlying host systems. Your organization's time, money, and values will usually determine which solution best fits your goals in this category.

Managed CI/CD Services

Managed services are completely hosted and supervised by an external organization offering CI/CD capabilities. The external organization takes responsibility for running and scaling their services, maintaining the health of their server fleet, and providing access to the services in a secure and easy to consume way. This is usually the largest value that managed CI/CD services provide and, in effect, can offload a large amount of work from your team by encapsulating and abstracting complexity.

While managed solutions delegate a portion of the work and allow your teams to focus on other concerns, there are areas where externalizing control over your infrastructure may not be a good idea. If compliance or your own organizational standards demand tight control over your processes, strictly regulated access to your code or data, or include special requirements that external partners are unable to guarantee, managing your own services might be your only option.

Self-Hosted CI/CD Services

If you are self-hosting your CI/CD systems, you will be responsible for making infrastructure decisions, keeping the underlying servers healthy by servicing hardware and patching software, and ensuring the services are available, secure, and performing adequately. This is a large scope of additional responsibility that may fall outside of your team's expertise and beyond what you're able to budget time for.

Because the CI/CD system has broad security access and is essential to developing trust in the changes you push to your projects, it is essential that your team treats it as a critical — rather than supplementary — component. Downtime of your continuous systems can affect the productivity and capabilities of your entire team. While the CI/CD system is not your main project and is meant to be a tool to help your organization, responsibility for managing it should not be taken lightly.

Security and Trust

Another major difference between using a managed provider and self-hosting your own continuous integration, delivery, and deployment service is the interplay between security and trust. Your CI/CD system will have access to your codebase and the ability to deploy to multiple environments. This means that you need to focus heavily on the security of your CI/CD systems. For managed solutions, you additionally need to evaluate your trust in the external provider.

Managed CI/CD Services

In many ways, managed CI/CD services make security much easier for most organizations. If you do not have any special regulatory requirements, most providers offer robust security that accesses your codebase through encrypted channels, runs your tests in isolated environments, etc. Since your provider is only concerned with CI/CD, they usually will have the focus and expertise to lock down the service adequately against threats like unauthorized access and accidental exposure of sensitive information. Their security footprint is well-defined and within their domain of proficiency.

On the other hand, when using a managed solution, much of your organization's security will come down to your faith in the capabilities and the integrity of your CI/CD provider. You are handing responsibility to an outside party, which decreases the burden on your team, but also relinquishes some level of accountability and control. For example, some drawbacks include the inability to implement any security mechanism that your provider does not support, reliance on your provider to patch high risk security vulnerabilities quickly,

and the requirement to trust that your provider is being candid about incidents and security reports they receive.

Self-Hosted CI/CD Services

Self-hosted CI/CD services have a different set of security challenges. The security of the entire system is now your own responsibility. This means that your teams will have to configure secure, isolated environments for your services and deployments, respond quickly to zero-day disclosures across all of the technologies involved in your CI/CD, implement strong access control mechanisms, and understand the complete extent of the system's security footprint. This can be an incredibly time consuming and difficult job that will likely require a full-time administrative or security team to administer properly. If you do not already have teams dedicated to those tasks for other parts of your infrastructure, it might be difficult to manage as your project grows.

However, managing your own service means that you also have a degree of flexibility and control over your security processes and tooling that wouldn't be possible otherwise. Your team can respond rapidly to threats instead of waiting for others to address problems. Your CI/CD infrastructure can be deployed securely within your own infrastructure behind multiple layers of protection. If you self-host your code repositories, you further minimize the surface area that any potential threats can target. Eliminating the need to go outside of your own private networks and secured perimeter to interact with such a critical tool helps minimize opportunities for lapses in security.

Integrations

Because CI/CD systems are a coordination point for different environments and many other pieces of software, it is important to consider how easily different solutions integrate with other tools or systems you are using. Each continuous integration system has a different set of projects they support natively. Some systems are built with plugin frameworks that allow users to create or use extensions to enhance the functionality or interoperability of the platform. These are all points to consider when choosing a CI/CD system.

Managed CI/CD Services

In general, managed CI/CD solutions typically provide a focused set of providers that they guarantee compatibility and support for. These first-class systems are often popular, well-known repository providers and developer tools that are reliable and predictable. Because of the relatively limited scope for integrations, the teams responsible for maintaining them are usually able to develop tight integrations and respond quickly to any dependency changes to ensure that functionality within the system does not suffer from breaking changes.

This reliable, tight integration with popular services works very well when it fits the requirements of your project. It can save your organization time and energy debugging compatibility issues, closely tracking dependency updates, and translating between systems that do not communicate natively. However, if your dependencies or software requirements fall outside of the set supported by a managed provider, there is generally little you can do to work around that issue. Your choices for other tools may start to be limited by the projects your CI/CD provider decides to support.

Self-Hosted CI/CD Services

Self-hosted services often have a much more varied landscape of projects they may interact with. Since most self-hosted solutions are open source, users and organizations can often influence the support that gets added to the system. Although not exclusively a feature of self-hosted services, a large number of projects have plugin systems with extensive libraries of integrations that help support many different projects and use-cases. The chances of finding an integration are much higher when the community can actively contribute to the development of the product and related components.

The downside of community involvement is that the quality of the integrations can vary significantly. Integration with popular services will likely be well-maintained and kept in good working order, but for any integration that falls outside of that scope, intensive testing may be required to ascertain the correctness and usability of the integration. These less popular integrations can become stagnant and break easily when changes are made to dependent projects or if they are not regularly updated to track the current versions of the CI/CD system. Furthermore, with community-provided plugins, it can be difficult to get support beyond filing a report on an issue tracker or eliciting informal peer-to-peer help through community forums or IRC channels. For critical integrations, your team may have to take responsibility for supporting and maintaining components internally.

Expense

Finally, one of the largest differences between managed CI/CD providers and self-hosting your own service is price. Managed services will almost always cost more up front due to the additional value being offered, but this can work out in different ways depending on the scale and requirements of your project and the solution you choose. You will have to analyze your current and future requirements to determine which costs might become untenable for your project down the line.

Managed CI/CD Services

Managed CI/CD services almost invariably have a higher cost than self-managed when looking at the basic numbers. With managed providers, you are hiring an outside organization to take care of a portion of the CI/CD work for you, so you can expect to pay for the work you are offloading. The cost though, can vary significantly. The pricing models will differ from service to service, with companies charging based on factors like number of users, number of concurrent jobs, type and scale of dedicated computing resources, number of projects, build minutes per month, and amount of parallelism, among others.

Frequently, managed providers start off fairly cheap, but can become more expensive quickly as your organization scales out. As your projects mature, you may suddenly find yourself having to use additional tiers of service that do not fit easily within your budget. When considering providers, it is important to take into account the costs now and anticipate what they may be in the future given different scenarios. Also, make note of the provider's pricing stability to minimize the chance of a pricing change down the line disrupting your plans. When analyzed thoroughly, understanding the pricing structure of the service you are using and budgeting proportionately can help you anticipate total operating costs much easier than you might be able to when self-hosting services.

Self-Hosted CI/CD Services

Self-hosted CI/CD systems are usually the cheaper option at first glance, but can have some hidden costs that should be assessed. Self-hosted options require computing resources to run, so initially, the largest cost is procuring the infrastructure that will run your continuous systems. This has become a cost effective option as infrastructure-as-a-service cloud computing platforms have increased in popularity. Self-hosted CI/CD tends to scale in costs more linearly than managed services as additional infrastructure can be provisioned and configured for a predictable cost as requirements change.

However, the price of infrastructure is only one component of the operational cost of running your own CI/CD services. By choosing to run your own systems, your team is taking on a significant amount of additional work and responsibility. As you scale out, the internal team responsible for managing your testing and deployment may need to grow and develop more complex processes. This has both direct costs in terms of hiring and training an internal staff and indirect opportunity costs incurred by having those individuals focus on managing your CI/CD systems instead of working on product features or other work. Overall, self-hosted services are typically much simpler to reason about in terms of infrastructure but more difficult in terms of total operational costs.

Is There a Middle Ground?

We've mentioned some of the differences between managed and self-hosted CI/CD services, but there are also some hybrid approaches that might be better suited for certain organizations. Some managed providers offer on-premises versions of their service as an alternative to their web-based software-as-a-service offerings. This can help bridge the gap between self-hosting and external management if your organization has requirements that make it difficult to interface with an external provider.

For example, if your security requirements dictate that your code not leave your internal network, on-premises deployments of paid or enterprise CI/CD solutions may be an option to consider. This can provide control over your network and the infrastructure used for the systems without sacrificing the support and accountability that an external contract may be able to guarantee. On-premises paid CI/CD can help address some of the potential challenges of using a managed service, but it is not a perfect solution. While this may be able to help with some of the compliance, security, and trust issues, it can be very expensive and will likely require your own staff to take on some of the management responsibilities. A hybrid solution takes on some of the benefits and some of the drawbacks of the other two categories.

Conclusion

Both managed CI/CD providers and self-hosted solutions can help improve your development and release practices to deliver more reliable products with more confidence. We've covered the most important differences between these two main options, some of the factors that may affect your decision, and the potential challenges you will have to overcome with each choice. We also talked briefly about offerings that attempt to straddle the line between self-hosting and managed solutions. In the end, your organization's unique requirements, budget, and administration bandwidth will help you decide which option offers the right tradeoffs.

By: Justin Ellingwood   ♡ Upvote (3)   ⊡ Subscribe   ⬆ Share

We just made it easier for you to deploy faster.

Related Tutorials

CI/CD Tools Comparison: Jenkins, GitLab CI, Buildbot, Drone, and Concourse

An Introduction to Continuous Integration, Delivery, and Deployment

How To Set Up Buildbot on FreeBSD

An Introduction to the Kubernetes DNS Service

Webinar Series: Building Blocks for Doing CI/CD with Kubernetes

0 Comments

Leave a comment...

Log In to Comment