AWS Elastic Beanstalk FAQs - Amazon Web Services (AWS)

aws.amazon.com (https://aws.amazon.com/elasticbeanstalk/faqs/)

General

Q: What is AWS Elastic Beanstalk?

AWS Elastic Beanstalk makes it even easier for developers to quickly deploy and manage applications in the AWS Cloud. Developers simply upload their application, and Elastic Beanstalk automatically handles the deployment details of capacity provisioning, load balancing, auto-scaling, and application health monitoring.

Q: Who should use AWS Elastic Beanstalk?

Those who want to deploy and manage their applications within minutes in the AWS Cloud. You don't need experience with cloud computing (https://aws.amazon.com/what-is-cloud-computing/) to get started. AWS Elastic Beanstalk supports Java, .NET, PHP, Node.js, Python, Ruby, Go, and Docker web applications.

Q: Which languages and development stacks does AWS Elastic Beanstalk support?

AWS Elastic Beanstalk supports the following languages and development stacks:

Apache Tomcat for Java applications

Apache HTTP Server for PHP applications

Apache HTTP Server for Python applications

Nginx or Apache HTTP Server for Node.js applications

Passenger or Puma for Ruby applications

Microsoft IIS 7.5, 8.0, and 8.5 for .NET applications

Java SE

Docker

Go

See Supported Platforms for a complete, up-to-date list of supported language and development stacks.

Q: Will AWS Elastic Beanstalk support other languages?

Yes. AWS Elastic Beanstalk is designed so that it can be extended to support multiple development stacks and programming languages in the future. AWS is working with solution providers on the APIs and capabilities needed to create additional Elastic Beanstalk offerings.

Q: What can developers now do with AWS Elastic Beanstalk that they could not before?

AWS Elastic Beanstalk automates the details of capacity provisioning, load balancing, auto scaling, and application deployment, creating an environment that runs a version of your application. You can simply upload your deployable code (e.g., WAR file), and AWS Elastic Beanstalk does the rest. The AWS Toolkit for Visual Studio and the AWS Toolkit for Eclipse allow you to deploy your application to AWS Elastic Beanstalk and manage it without leaving your IDE. Once your application is running, Elastic Beanstalk automates management tasks–such as monitoring, application version deployment, a basic health check–and facilitates log file access. By using Elastic Beanstalk, developers can focus on developing their application and are freed from deployment-oriented tasks, such as provisioning servers, setting up load balancing, or managing scaling.

Q: How is AWS Elastic Beanstalk different from existing application containers or platform-as-a-service solutions?

Most existing application containers or platform-as-a-service solutions, while reducing the amount of programming required, significantly diminish developers' flexibility and control. Developers are forced to live with all the decisions predetermined by the vendor–with little to no opportunity to take back control over various parts of their application's infrastructure. However, with AWS Elastic Beanstalk, developers retain full control over the AWS resources powering their application. If developers decide they want to manage some (or all) of the elements of their infrastructure, they can do so seamlessly by using Elastic Beanstalk's management capabilities.

Q: What elements of my application can I control when using AWS Elastic Beanstalk?

With AWS Elastic Beanstalk, you can:

Select the operating system that matches your application requirements (e.g., Amazon Linux or Windows Server 2012 R2)

Choose from several available database and storage options

Enable login access to Amazon EC2 instances for immediate and direct troubleshooting

Quickly improve application reliability by running in more than one Availability Zone

Enhance application security by enabling HTTPS protocol on the load balancer

Access built-in Amazon CloudWatch monitoring and getting notifications on application health and other important events

Adjust application server settings (e.g., JVM settings) and pass environment variables

Run other application components, such as a memory caching service, side-by-side in Amazon EC2

Access log files without logging in to the application servers

Q: What are the Cloud resources powering my AWS Elastic Beanstalk application?

AWS Elastic Beanstalk uses proven AWS features and services, such as Amazon EC2, Amazon RDS, Elastic Load Balancing, Auto Scaling, Amazon S3, and Amazon SNS, to create an environment that runs your application. The current version of AWS Elastic Beanstalk uses the Amazon Linux AMI or the Windows Server 2012 R2 AMI.

Q: What kinds of applications are supported by AWS Elastic Beanstalk?

AWS Elastic Beanstalk supports Java, .NET, PHP, Node.js, Python, Ruby, Go, and Docker, and is ideal for web applications. However, due to Elastic Beanstalk's open architecture, non-web applications can also be deployed using Elastic Beanstalk. We expect to support additional application types and programming languages in the future. See Supported Platforms (https://docs.aws.amazon.com/elasticbeanstalk/latest/dg/concepts.platforms.html) to learn more.

Q: Which operating systems does AWS Elastic Beanstalk use?

AWS Elastic Beanstalk runs on the Amazon Linux AMI and the Windows Server 2012 R2 AMI. Both AMIs are supported and maintained by Amazon Web Services and are designed to provide a stable, secure, and high-performance execution environment for Amazon EC2 Cloud computing.

Getting Started

Q: How do I sign up for AWS Elastic Beanstalk?

To sign up for AWS Elastic Beanstalk, choose the Sign Up Now button on the Elastic Beanstalk detail page. You must have an Amazon Web Services account to access this service; if you do not already have one, you will be prompted to create one when you begin the Elastic Beanstalk process. After signing up, please refer to the AWS Elastic Beanstalk Getting Started Guide.

Q: Why am I asked to verify my phone number when signing up for AWS Elastic Beanstalk?

AWS Elastic Beanstalk registration requires you to have a valid phone number and email address on file with AWS in

case we ever need to contact you. Verifying your phone number takes only a few minutes and involves receiving an automated phone call during the registration process and entering a PIN number using the phone key pad.

Q: How do I get started after I have signed up?

The best way to get started with AWS Elastic Beanstalk is to work through the AWS Elastic Beanstalk Getting Started Guide, part of our technical documentation. Within a few minutes, you will be able to deploy and use a sample application or upload your own application.

Q: Is there a sample application that I can use to check out AWS Elastic Beanstalk?

Yes. AWS Elastic Beanstalk includes a sample application that you can use to test drive the offering and explore its functionality.

Databases and Storage

Q: Does AWS Elastic Beanstalk store anything in Amazon S3?

Yes. AWS Elastic Beanstalk stores your application files and, optionally, server log files in Amazon S3. If you are using the AWS Management Console, the AWS Toolkit for Visual Studio, or AWS Toolkit for Eclipse, an Amazon S3 bucket will be created in your account for you and the files you upload will be automatically copied from your local client to Amazon S3. Optionally, you may configure Elastic Beanstalk to copy your server log files every hour to Amazon S3. You do this by editing the environment configuration settings.

Q: Can I use Amazon S3 to store application data, like images?

Yes. You can use Amazon S3 for application storage. The easiest way to do this is by including the AWS SDK as part of your application's deployable file. For example, you can include the AWS SDK for Java as part of your application's WAR file.

Q: What database solutions can I use with AWS Elastic Beanstalk?

AWS Elastic Beanstalk does not restrict you to any specific data persistence technology. You can choose to use Amazon Relational Database Service (Amazon RDS) or Amazon DynamoDB, or use Microsoft SQL Server, Oracle, or other relational databases running on Amazon EC2.

Q: How do I set up a database for use with AWS Elastic Beanstalk?

Elastic Beanstalk can automatically provision an Amazon RDS DB instance. The information about connectivity to the DB instance is exposed to your application by environment variables. To learn more about how to configure RDS DB instances for your environment, see the Elastic Beanstalk Developer Guide.

Q: Does this mean I need to modify the application code when moving from test to production?

Not with AWS Elastic Beanstalk. With Elastic Beanstalk, you can specify the connection information in the environment configuration. By extracting the connection string from the application code, you can easily configure different Elastic Beanstalk environments to use different databases.

Security

Q: How do I make my application private?

By default, your application is available publicly at myapp.elasticbeanstalk.com for anyone to access. You can use Amazon VPC to provision a private, isolated section of your application in a virtual network that you define. This virtual network can be made private through specific security group rules, network ACLs, and custom route tables. You can also easily control what other incoming traffic, such as SSH, is delivered or not to your application servers by changing the EC2 security group settings.

Q: Can I run my application inside a Virtual Private Cloud (VPC)?

Yes, you can run your applications in a VPC. For more details, see the AWS Elastic Beanstalk Developer Guide.

Q: Where can I find more information about security and running applications on AWS?

For more information about security on AWS, please refer to our Amazon Web Services: Overview of Security Processes document and visit our Security Center.

Q: Is it possible to use Identity & Access Management (IAM) with AWS Elastic Beanstalk?

Yes. IAM users with the appropriate permissions can now interact with AWS Elastic Beanstalk.

Q: Why should I use IAM with AWS Elastic Beanstalk?

IAM allows you to manage users and groups in a centralized manner. You can control which IAM users have access to AWS Elastic Beanstalk, and limit permissions to read-only access to Elastic Beanstalk for operators who should not be able to perform actions against Elastic Beanstalk resources. All user activity within your account will be aggregated under a single AWS bill.

Q: How do I create IAM users?

You can use the IAM console, IAM command line interface (CLI), or IAM API to provision IAM users. By default, IAM users have no access to AWS services until permissions are granted.

Q: How do I grant an IAM user access to AWS Elastic Beanstalk?

You can grant IAM users access to services by using policies. To simplify the process of granting access to AWS Elastic Beanstalk, you can use one of the policy templates in the IAM console to help you get started. Elastic Beanstalk offers two templates: a read-only access template and a full-access template. The read-only template grants read access to Elastic Beanstalk resources. The full-access template grants full access to all Elastic Beanstalk operations, as well as permissions to manage dependent resources, such as Elastic Load Balancing, Auto Scaling, and Amazon S3. You can also use the AWS Policy Generator to create custom policies. For more details, see the AWS Elastic Beanstalk Developer Guide.

Q: Can I restrict access to specific AWS Elastic Beanstalk resources?

Yes. You can allow or deny permissions to specific AWS Elastic Beanstalk resources, such as applications, application versions, and environments.

Q: Who gets billed for the AWS resources that an IAM user creates?

All resources created by IAM users under a root account are owned and billed to the root account.

Q: Who has access to an AWS Elastic Beanstalk environment launched by an IAM user?

The root account has full access to all AWS Elastic Beanstalk environments launched by any IAM user under that account. If you use the Elastic Beanstalk template to grant read-only access to an IAM user, that user will be able to view all applications, application versions, environments, and any associated resources in that account. If you use the Elastic Beanstalk template to grant full access to an IAM user, that user will be able to create, modify, and terminate any Elastic Beanstalk resources under that account.

Q: Can an IAM user access the AWS Elastic Beanstalk console?

Yes. An IAM user can access the AWS Elastic Beanstalk console using their username and password.

Q: Can an IAM user call the AWS Elastic Beanstalk API?

Yes. An IAM user can use their access key and secret key to perform operations using the Elastic Beanstalk API.

Q: Can an IAM user use the AWS Elastic Beanstalk command line interface?

Yes. An IAM user can use their access key and secret key to perform operations using the AWS Elastic Beanstalk command line interface (CLI).

Managed Platform Updates

Q: How can I keep the underlying platform of the environment running my application automatically up-to-date?

You can opt-in to having your AWS Elastic Beanstalk environments automatically updated to the latest version of the

underlying platform running your application during a specified maintenance window. Elastic Beanstalk regularly releases new versions of supported platforms (Java, PHP, Ruby, Node.js, Python, .NET, Go, and Docker) with operating system, web and application server, and language and framework updates.

Q: How can I get started with managed platform updates?

To let Elastic Beanstalk automatically manage your platform updates, you must enable managed platform updates in the Configuration tab of the Elastic Beanstalk console or use the EB CLI or API. After you have enabled the feature, you can configure which types of updates to allow and when updates can occur.

Q: What kinds of platform version updates will managed platform updates apply?

AWS Elastic Beanstalk can automatically perform platform updates for new patch and minor platform versions. Elastic Beanstalk will not automatically perform major platform version updates (e.g., Java 7 Tomcat 7 to Java 8 Tomcat 8) because they include backwards incompatible changes and require additional testing. In these cases, you must manually initiate the update.

Q: How does AWS Elastic Beanstalk distinguish between "major," "minor," and "patch" version releases?

AWS Elastic Beanstalk platforms are versioned using this pattern: MAJOR.MINOR.PATCH (e.g., 2.0.0). Each portion is incremented as follows:

MAJOR version when there are incompatible changes.

MINOR version when there is additional functionality added in a backward-compatible manner.

PATCH version when there are backward-compatible bug fixes.

Q: When and how can I perform major version updates?

You can perform major version updates at any time using the AWS Elastic Beanstalk management console, API, or CLI. You have the following options to perform a major version update:

Apply the update in-place on an existing environment. See Updating Your Elastic Beanstalk Environment's Platform Version.

Create a clone of an existing environment with the new platform version. See Clone an Environment to learn more.

Q: How does Elastic Beanstalk apply managed platform updates?

The updates are applied using an immutable deployment mechanism that ensures that no changes are made to the existing environment until a parallel fleet of Amazon EC2 instances, with the updates installed, is ready to be swapped with the existing instances, which are then terminated. In addition, if the Elastic Beanstalk health system

detects any issues during the update, traffic is redirected to the existing fleet of instances, ensuring minimal impact to end users of your application.

Q: Will my application be available during the maintenance windows?

Since managed platform updates use an immutable deployment mechanism to perform the updates, your application will be available during the maintenance window and consumers of your application will not be impacted by the update.

Q: What does it cost to use managed platform updates?

There is no additional charge for the managed platform updates feature. You simply pay for the additional EC2 instances necessary to perform the update for the duration of the update.

Q: What is a maintenance window?

A maintenance window is a weekly two-hour-long time slot during which AWS Elastic Beanstalk will initiate platform updates if managed platform updates is enabled and a new version of the platform is available. For example, if you select a maintenance window that begins every Sunday at 2 AM, AWS Elastic Beanstalk will initiate the platform update sometime between 2-4 AM every Sunday. It is important to note that, depending on the configuration of your applications, updates could complete outside of the maintenance window.

The maintenance window is set on a per-environment basis, providing you the option to set different maintenance windows for your various application components or applications. This allows environment updates to be staggered if you do not want multiple pieces of your application to be updated at the same time. If you enable managed platform updates but do not specify a maintenance window, a default weekly 2-hour window will be assigned for your environment. If you want to change when maintenance is performed on your behalf, you can do so by modifying the managed update configuration in the AWS Management Console or by using the UpdateEnvironment API.

Q: How will I be notified of the availability of new platform versions?

You will be notified about the availability of new platform versions through the AWS Management Console, forum announcements, and release notes.

Q: Where can I find details of changes between platform versions?

Details on changes between platform versions can be found on the AWS Elastic Beanstalk Release Notes page.

Q: What operations can I perform on the environment while a managed update is in progress?

The only action available to you while a managed platform update is in-progress is 'abort'. This will allow you to stop the update immediately and roll back to the previous version.

Q: Which platform version will my environment be updated to if there are multiple new versions released in between maintenance windows?

Your environment will always be updated to the latest version available based on the level (minor plus patch or patch only) you have selected.

Q: Where can I find details of all the managed platform updates that have been performed on my environment?

Details for every managed platform update are available on the events page and are tagged with an event type of "MAINTENANCE."

Q: How often are platform version updates released?

The number of version releases in a given year varies based on the frequency and content of releases and patches from the language/framework's vendor or core team, and the outcome of a thorough vetting of these releases and patches by our platform engineering team.

Billing

Q: How much does AWS Elastic Beanstalk cost?

There is no additional charge for AWS Elastic Beanstalk–you pay only for the AWS resources actually used to store and run your application.

Q: How much do the AWS resources powering my application on AWS Elastic Beanstalk cost?

You pay only for what you use, and there is no minimum fee for the use of any AWS resources. For Amazon EC2 pricing information, please visit the pricing section on the EC2 detail page. For Amazon S3 pricing information, please visit the pricing section on the S3 detail page. You can use the AWS simple calculator to estimate your bill for different application sizes.

Q: How do I check how many AWS resources have been used by my application and access my bill?

You can view your charges for the current billing period at any time on the Amazon Web Services web site by logging into your Amazon Web Services account and choosing Account Activity under Your Web Services Account.

Support

Q: Does AWS Support cover AWS Elastic Beanstalk?

Yes. AWS Support covers issues related to your use of AWS Elastic Beanstalk. For further details and pricing, see the AWS Support page.

Q: What other support options are available?

You can tap into the breadth of existing AWS community knowledge to help you with your development through the

AWS Elastic Beanstalk discussion forum.

Learn more about AWS Elastic Beanstalk Pricing

Visit the AWS Elastic Beanstalk Pricing Page

Ready to get started?

Sign up (https://portal.aws.amazon.com/gp/aws/developer/registration/index.html)

Have more questions?

Contact us (https://aws.amazon.com/contact-us/)

Page Content

General Getting Started Databases and Storage Security Managed Platform Updates Billing Support

aws.amazon.com (https://aws.amazon.com/elasticbeanstalk/faqs/)