

Alibaba Cloud Web App Service

Managing Environments

Issue: 20200417









Legal disclaimer

Alibaba Cloud reminds you to carefully read and fully understand the terms and conditions of this legal disclaimer before you read or use this document. If you have read or used this document, it shall be deemed as your total acceptance of this legal disclaimer.

- 1.** You shall download and obtain this document from the Alibaba Cloud website or other Alibaba Cloud-authorized channels, and use this document for your own legal business activities only. The content of this document is considered confidential information of Alibaba Cloud. You shall strictly abide by the confidentiality obligations. No part of this document shall be disclosed or provided to any third party for use without the prior written consent of Alibaba Cloud.
- 2.** No part of this document shall be excerpted, translated, reproduced, transmitted, or disseminated by any organization, company, or individual in any form or by any means without the prior written consent of Alibaba Cloud.
- 3.** The content of this document may be changed due to product version upgrades, adjustments, or other reasons. Alibaba Cloud reserves the right to modify the content of this document without notice and the updated versions of this document will be occasionally released through Alibaba Cloud-authorized channels. You shall pay attention to the version changes of this document as they occur and download and obtain the most up-to-date version of this document from Alibaba Cloud-authorized channels.
- 4.** This document serves only as a reference guide for your use of Alibaba Cloud products and services. Alibaba Cloud provides the document in the context that Alibaba Cloud products and services are provided on an "as is", "with all faults" and "as available" basis. Alibaba Cloud makes every effort to provide relevant operational guidance based on existing technologies. However, Alibaba Cloud hereby makes a clear statement that it in no way guarantees the accuracy, integrity, applicability, and reliability of the content of this document, either explicitly or implicitly. Alibaba Cloud shall not bear any liability for any errors or financial losses incurred by any organizations, companies, or individuals arising from their download, use, or trust in this document. Alibaba Cloud shall not, under any circumstances, bear responsibility for any indirect, consequential, exemplary, incidental, special, or punitive damages, including lost profits arising from the use or trust in this document, even if Alibaba Cloud has been notified of the possibility of such a loss.

- 5.** By law, all the contents in Alibaba Cloud documents, including but not limited to pictures, architecture design, page layout, and text description, are intellectual property of Alibaba Cloud and/or its affiliates. This intellectual property includes, but is not limited to, trademark rights, patent rights, copyrights, and trade secrets. No part of this document shall be used, modified, reproduced, publicly transmitted, changed, disseminated, distributed, or published without the prior written consent of Alibaba Cloud and/or its affiliates. The names owned by Alibaba Cloud shall not be used, published, or reproduced for marketing, advertising, promotion, or other purposes without the prior written consent of Alibaba Cloud. The names owned by Alibaba Cloud include, but are not limited to, "Alibaba Cloud", "Aliyun", "HiChina", and other brands of Alibaba Cloud and/or its affiliates, which appear separately or in combination, as well as the auxiliary signs and patterns of the preceding brands, or anything similar to the company names, trade names, trademarks, product or service names, domain names, patterns, logos, marks, signs, or special descriptions that third parties identify as Alibaba Cloud and/or its affiliates.
- 6.** Please contact Alibaba Cloud directly if you discover any errors in this document.

Document conventions

Style	Description	Example
	A danger notice indicates a situation that will cause major system changes, faults, physical injuries, and other adverse results.	 Danger: Resetting will result in the loss of user configuration data.
	A warning notice indicates a situation that may cause major system changes, faults, physical injuries, and other adverse results.	 Warning: Restarting will cause business interruption. About 10 minutes are required to restart an instance.
	A caution notice indicates warning information, supplementary instructions, and other content that the user must understand.	 Notice: If the weight is set to 0, the server no longer receives new requests.
	A note indicates supplemental instructions, best practices, tips, and other content.	 Note: You can use Ctrl + A to select all files.
>	Closing angle brackets are used to indicate a multi-level menu cascade.	Click Settings > Network > Set network type.
Bold	Bold formatting is used for buttons, menus, page names, and other UI elements.	Click OK.
Courier font	Courier font is used for commands.	Run the <code>cd /d C:/window</code> command to enter the Windows system folder.
Italic	Italic formatting is used for parameters and variables.	<code>bae log list --instanceid Instance_ID</code>
[] or [a b]	This format is used for an optional value, where only one item can be selected.	<code>ipconfig [-all -t]</code>

Style	Description	Example
{ } or {a b}	This format is used for a required value, where only one item can be selected.	switch {active stand}

Contents

Legal disclaimer.....	1
Document conventions.....	1
1 Overview of environment details.....	1
2 Configure a deployment environment.....	5
2.1 Deployment environment overview.....	5
2.2 Reverse proxy.....	6
2.3 Virtual Private Cloud (VPC).....	7
2.4 Configure ECS Instances.....	8
2.5 Server Load Balancer.....	13
2.6 ApsaraDB for RDS instances.....	17
2.7 ApsaraDB for Redis instances.....	24
2.8 Configure a service port.....	30
2.9 Configure health checks.....	30
2.10 Commands and lifecycle hooks.....	33
2.11 Environment variables.....	36
2.12 Configure parameters for a Java virtual machine.....	39
3 Configure HTTPS for a deployment environment.....	41
4 View metrics for a environment.....	44
5 View events.....	45
6 View logs.....	46

1 Overview of environment details

On the Environment Details page of the Web App Service console, you can perform common operations on an environment. The operations include start, stop, restart, release, and delete. You can also create a configuration template, upgrade or downgrade a technology stack, and download a Wpfile. On the Environment Details page, you can view specific information, such as the environment version, status, technology stack, creator, public IP address, latest events, and resources.

Environment details

The Environment Details page includes specific information, such as the package version, technology stack, creator, creation time, latest events, and resources.

Package Version

The package version of an application that runs in an environment. You can click a package version to download the package for an environment.

Status

The status of an environment. You can move the pointer over the dot next to the name of an environment to view the status of the environment. Available statuses include Initializing, Running, Starting, Restarting, Stopping, Stopped, Releasing, Released, Abnormal, and Locked.

Tech Stack


The version of the operating system that runs in an environment, the name of the platform, and the version of the architecture.

Public Address

The IP address that is accessible by clients over the Internet. If no Server Load Balancer (SLB) instance is bound with an environment, the IP address of an ECS instance is displayed in the field. If an SLB instance is bound with the environment, the IP address of the SLB instance is displayed in the field.

Latest Events

The latest events that occur in an environment. When you perform management tasks on an environment, Web App Service provides notifications that describe the latest events.

These events allow you to track environment changes. You can click the  icon to open the **Events** page that shows all the event history for an environment.

Resources

On the **Resources** tab, resources that reside in an environment are displayed. On the Resources tab, available resources, such as VPCs and ECS instances, are listed.

On the Resources tab, you can perform the following operations on an ECS instance:

- Click the ID of the ECS instance to go to the Instance Details page.
- Click the public IP address of the ECS instance to view the homepage of an application.
- Click **Remote Connection** to connect to the ECS instance.
- Click **Switch to Subscription** to open the ECS console and change the billing method of the ECS instance.

Deploy an environment

In the upper-right corner of the **Overview** page, click **Deploy**. In the **Environment** dialog box, select a new package to update the environment.

1. Select a **Package Source**:

- **Upload Local Application:** Click **Select File** to upload a local package, and configure the **Package Version** again. In the **Version Description** field, enter an informative description to identify this operation.
- **Select an earlier version.:** Select a historical version to deploy the environment again.

2. Click **OK** to complete the configurations.

Start and stop an environment

Stop an environment

Ensure that an environment is in the Running state. Then, in the upper-right corner of the **Overview** page, click **Stop** to stop the environment. If an environment is stopped, no charge is incurred for applications that run in the environment. However, you are still charged for resources, such as ECS instances and SLB instances, that reside in the environment.

Start an environment

Ensure that an environment is in the Stopped state. Then, in the upper-right corner of the **Overview** page, click **Start** to start the environment. All settings in the Applications section of the Configurations page are applied after an environment is started.

Restart an environment

In the upper-right corner of the **Overview** page for an environment, click **Restart** to restart the environment. A restart operation for an environment will not terminate or restart resources that reside in the environment. If errors occur when an environment responds to some unexpected requests, a restart operation may help you fix these errors and identify the causes of these errors.

Rebuild an environment

Ensure that an environment is in the Abnormal state. Then, in the upper-right corner of the **Overview** page, click **Rebuild** to rebuild the environment. All resources that reside in the environment are built again. This operation can help you recover the environment and identify the cause of an error.

Release an environment

In the upper-right corner of the **Overview** page for an environment, click **Release** to release all resources that reside in the environment and delete the environment for an application. No charges are incurred for released resources, such as ECS instances and SLB instances. You will only be charged a minimum amount for storage.

Delete an environment

In the upper-right corner of the **Overview** page for an environment, click **Delete** to delete the environment for the application. Web App Service will delete all ECS instances, SLB instances, and environment metadata that resides in the environment.




Notice:

When releasing the environment, Web+ will not release the following types of resources, please confirm whether you need it and release it manually:

- Database products: RDS and Redis.
- Resources in Subscription billing method.
- Resources imported from elsewhere.

Create a configuration template


1. In the upper-right corner of the **Overview** page for an environment, choose  > **Create Configuration Template**.
2. In the **Create Configuration Template** dialog box, enter the name and description of a configuration template, and click **OK**.

Web App Service creates a new configuration template based on the environment. You can use the configuration template to create new deployment environments.

In the **Create Configuration Template** dialog box, click **View the new configuration template** to open the **Templates** page.


Upgrade or downgrade a technology stack

During the downgrade or upgrade process for a technology stack, Web App Service updates the technology stack of an environment. Then, the application that runs in the environment will be deployed again. Before you upgrade or downgrade a technology stack, we recommend that you evaluate the compatibility between the target technology stack version and the application.

1. In the upper-right corner of the **Overview** page, choose  > **Upgrade/Downgrade Technology Stack**.
2. In the **Upgrade/Downgrade Technology Stack** dialog box, select the target **Tech Stack Version**, and click **OK**.

Download a Wpfile file

You can modify a Wpfile and use the **apply** command in the CLI tool to update or create an environment after you download a Wpfile.

1. In the upper-right corner of the **Overview** page, choose  > **Download WP File**.
2. Open the Wpfile file on a localhost to view and change the configurations of the environment.

2 Configure a deployment environment

2.1 Deployment environment overview

Web App Service allows you to customize deployment environments. When you create an application or deployment environment, Web App Service provisions one or more resources to run these applications. These resources are listed as follows.

- [Reverse proxy](#): Web App Service has a built-in reverse proxy. After the reverse proxy is started, it listens on port 80 of ECS instances and relays HTTP requests to applications that run on these ECS instances.
- [Virtual Private Cloud \(VPC\)](#): A Virtual Private Cloud (VPC) is a dedicated private network on the cloud. VPCs allow you to securely route traffic to applications that run on ECS instances in Web App Service. If no VPC is configured when you start a deployment environment, Web App Service uses the default VPC.
- [Configure ECS Instances](#): You can run applications for the specified platforms of technology stacks on ECS instances. You must configure settings, such as the instance type, number of instances, security group, key pair, and whether to enable an Internet IP address.
- [Server Load Balancer](#): With traffic distribution, SLB allows you to enhance service capabilities of applications, remove single points of failure, and improve the availability of applications.
- [ApsaraDB for RDS instances](#): Web App Service allows you to orchestrate ApsaraDB for RDS resources. After configuring settings, such as the database type, database version, and storage type, you can add ApsaraDB for RDS instances to your deployment environment.
- [Configure a service port](#): In Web App Service, the service port of an application is open on an ECS instance. After you configure a service port, Web App Service routes requests to the service port of the specified application. This allows clients to access the application.
- [Configure health checks](#): With health checks, Web App Service allows you to check whether the status of running applications is healthy. After you create health checks, Web App Service will perform corresponding operations on applications based on the health check status. For example, Web App Service restarts an ECS instance if a health

check for the ECS instance fails. It also restarts an application instance if the health check status of the application instance is abnormal.

- [Commands and lifecycle hooks](#): Web App Service provides features, such as customizing start commands, stop commands, and lifecycle hooks, to extend service capabilities. These features help you change default management tasks to ensure that they can meet your business requirements.
- [Environment variables](#): You can specify a set of environment variables for each application. After you deploy an application, you can use these environment variables to change the application settings. Environment variables specified for each application are exclusive. You can specify the same environment variable for multiple applications but the value of the environment variable can be different for each application.
- [Configure parameters for a Java virtual machine](#): JVM settings are used to configure containers when you start an application. Accurate configuration of these settings helps you reduce the overhead of garbage collection (GC), decrease server response time, and improve overall throughput.

2.2 Reverse proxy

Web+ has a built-in reverse proxy. After the reverse proxy is started, it listens on port 80 of ECS instances. It also replays HTTP requests that reach port 80 to the service port of applications that run on these ECS instances.

Procedure

In Web+, you can use **NGINX** as a reverse proxy and open port 80 for access to an application from clients.

1. Log on to the [Web+ console](#).
2. On the **Overview** page, click **View All** in the upper-right corner of the **Last Updated Deployment Environments** section.
3. On the **Applications and Deployment Environments** page, click the > icon next to the name of an application to show a list of linked deployment environments.



Note:

On the **Deployment Environments** list, the most recently updated four deployment environments are displayed. If the required deployment environment is displayed on the list, you can click the specific name to go to the Deployment Environment Details page.

4. Click the specific name of a deployment environment to go to the **Deployment Environment Details** page.
5. In the left-side navigation pane, click **Configurations**.
6. In the **Platforms** section, you can view the **Reverse Proxy** setting, turn on the **Enable Reverse Proxy** switch, and select **Nginx (1.14.2)** in the Reverse Proxy Type section.
7. After the configuration is complete, click **Change Configuration** to enable the configuration.

2.3 Virtual Private Cloud (VPC)

A Virtual Private Cloud (VPC) is a logically isolated private network on the cloud. With VPCs, traffic can be securely routed to Elastic Compute Service (ECS) instances that run in Web+. If no VPC is configured when you start an environment, Web+ uses the [default VPC](#). You can start an environment in a custom VPC to configure network and security settings. In Web+, you can select a subnet for your resources. Then, you need to configure IP addresses for instances and load balancers in your environment. When you create an environment, a VPC is specified by default. You can change the settings for subnets and IP addresses that are related to your environment.

Step 1: Create a VPC and a VSwitch

To deploy cloud resources in a VPC, you must first plan the network, create a VPC and create at least one VSwitch. To plan the network, see [Network planning](#).

To create a VPC and a VSwitch, follow these steps:

1. Log on to the [VPC console](#).
2. Select a region.

The VPC and the cloud resources to deploy must be in the same region. In this tutorial, select the **China (Qingdao)** region.

3. Configure the VPC and VSwitch. Descriptions about the configuration items are provided in the following table.

**Note:**

In this tutorial, do not select **Assign IPv6 Address Free of Charge**.


Step 2: Configure a VPC in the Web+ console

1. Log on to the [Web+ console](#).

2. On the **Overview** page, click **View All** in the upper-right corner of the **Last Updated Environments** section.
3. On the **Applications and Environments** page, click the > icon next to the name of an application to show a list of linked deployment environments.

**Note:**

The **Environments** list shows the last updated four deployment environments. If the required deployment environment is listed, you can click the name of the deployment environment to go to the Environment Details page.

4. Click the name of an environment to open the **Overview** page of the environment.
5. In the left-side navigation pane, click **Configurations** to open the Configurations page.
6. In the **Environment** section, click **Network**, and click  to update the VPC setting for the environment that resides in Web+.
7. On the **VPC** drop-down list, select a VPC, and select one or more VSwitches on the **VSwitch** drop-down list.

**Note:**

Web+ applications must be deployed in a private network that is managed by a VSwitch. You can deploy applications to zones that are managed by different VSwitches to improve availability of the applications.

8. In the upper-right corner, click **Change Configuration** to apply the update after the configuration is complete.

2.4 Configure ECS Instances

When you create or modify an environment, Web+ creates one or more ECS instances to run applications on your specified platforms. In the Web+ console, you can change settings for ECS instances that reside in your environment.

Background information

Elastic Compute Service (ECS) is a Web service that provides elastically scalable compute capacity for Alibaba Cloud. In Web+, the billing method of new ECS instances is pay-as-you-go by default. You can log on to the [ECS console](#) to change the billing method to subscription for these ECS instances that are created in Web+. When you release an environment or scale down ECS instances, Web+ will not stop or release subscription ECS instances. You need to manually release these subscription ECS instances.

In most cases, Web+ releases ECS instances by default when you release an environment or scale down ECS instances. You can use the CLI tool to enable or disable the no fees for stopped instances feature for ECS instances. After an ECS instance is stopped, no charge is incurred for using vCPU, memory, and Internet IP address in stop mode. However, you are only charged a minimum expense for disks that are retained. You can log on to the [ECS console](#) to back up data and release all disks.

When you create or change an environment, you can change the settings for ECS instances. When you modify ECS settings, Web+ will update the current ECS settings or replace the current ECS instances with new ECS instances to meet your business requirements. In most cases, configuration changes will cause the current ECS settings to be updated. However, if either of the following configuration changes is included, Web+ will replace the current ECS instances with new ECS instances. For example, Web+ will replace the current ECS instances with new ECS instances.

- Instance type changes
- System disk scale-down

Open the ECS Instance Settings page

When you create an environment or change settings for a running environment, you can change ECS settings.

Change ECS settings when creating applications or environments

1. Log on to the [Web+ console](#).
2. On the **Overview** page, click **Create** in the upper-right corner of the **Last Updated Environments** section. Follow the instructions to configure the required settings in the **Basic Information** and **Environment Information** steps and go to the **Configurations** step.
3. In the **Configurations** step, select **Custom** in the **Pre-defined Configuration** setting.
4. On the list of displayed settings, select **Instances** in the Resources section and configure the required settings.
5. Click **Creation Complete** to create a new environment. The ECS settings will also take effect.

Change ECS settings when creating a new environment for an existing application

1. Log on to the [Web+ console](#).

2. On the **Overview** page, click **View All** in the upper-right corner of the **Last Updated Environments** section.
3. On the **Applications and Environments**, click the ID of an application for which you want to create a new environment.
4. On the **Application Details** page, click **Create Environment**, configure the required settings in the Environment Information step, and go to the **Configurations** page.
5. In the **Configurations** step, select **Custom** in the **Pre-defined Configuration** setting.
6. On the list of displayed settings, select **Instances** and configure the required settings.
7. Click **Creation Complete** to create a new environment.

Change ECS settings when modifying an environment

1. Log on to the [Web+ console](#).
2. On the **Overview** page, click **View All** in the upper-right corner of the **Last Updated Environments** section.
3. On the **Applications and Environments** page, click the > icon next to the name of an application to view a list of linked environments.



Note:

On the **Environments** list, the most recently updated four environments are displayed. If the required environment is displayed on the list, you can click the specific name to go to the Environment Details page.

4. Click the specific name of an environment to go to the **Environment Details** page.
5. In the left-side navigation pane, click **Configurations** to go to the Configurations page.
6. On the list of displayed settings, select **Instances** in the Resources section and configure the required settings.
7. Click **Change Configuration** to update the environment. The ECS settings will also take effect.

ECS settings

Instance Type

You can select a maximum of 10 instance types. Web+ creates ECS instances of specified instance types in a specific order or priority. The priority of each instance type is based on the chronological order when an instance type is selected. The first selected instance type is of the highest priority. If an instance type of top priority runs out of stock, Web+ uses another instance type of the next priority level to create ECS instances. After you remove an

instance type from the environment configurations, to replace the existing ECS instances of this type, Web+ will create ECS instances of another selected type.

Instances

For each environment, the number of instances that you can create ranges from 0 to 100. If the specified number is less than the number of existing instances, Web+ stops and releases redundant ECS instances. If the specified number is more than the number of existing instances, Web+ creates new ECS instances.

When you release an environment or scale down ECS instances, Web+ will release ECS instances by default. You can use the CLI tool to enable or disable the no fees for stopped instances feature for ECS instances. After an ECS instance is stopped, no charge is incurred for using vCPU, memory, and Internet IP address in this stop mode. However, you are only charged a minimum expense for disks that are retained. Web+ allows you to release an environment to manage ECS instances. When you release an environment or scale down ECS instances, Web+ releases ECS instances. In addition to the console, you can use the CLI tool to release ECS instances.

For large-scale and availability-sensitive applications, we recommend that you create at least two ECS instances for backup. If any errors occur on an ECS instance, services can be switched over to another ECS instance. The redundant ECS instances will help ensure business continuity.

Security Group

For each environment, Web+ will create a security group. By default, port 80 and port 22 are open to allow access through the Internet. You can change the rules of the security group. For example, you can limit access to port 22 from specified IP addresses over the Internet. After the default security group is created, Web+ retains the default rules. Therefore, we recommend that you guarantee business continuity after you change the rules for the default security group.

In addition to the default security group, you can also configure other security groups. After you submit changes, Web+ adds existing ECS instances to the specified security groups. New ECS instances will be added to both the default security group and the specified security groups.

Key Pair

If you want to use Secure Shell (SSH) to access ECS instances that are created by Web+, you can configure a key pair. After you change a key pair and submit the change, Web+ will update the key pair. After the update is complete, you need to restart ECS instances to enable the key pair.

System Disk Size

By default, the type of system disk for ECS instances that Web+ creates is Enhanced SSD with a disk size of 100 GB. You can change the size of a system disk and the size ranges from 40 GB to 500 GB. If you scale up a system disk, Web+ changes the size of the system disk for the specified ECS instance. If you scale down a system disk, Web+ creates a new system disk for the specified ECS instance to replace the existing system disk.

Enable Internet IP Address

If you enable an Internet IP address, Web+ creates each ECS instance with an Internet IP address. Otherwise, Web+ creates each ECS instance that is not accessible through the Internet.

Use the CLI tool to configure ECS instances

The CLI tool uses the Wpfile file to configure ECS instances. You can use the **wpctl dump** command to retrieve the Wpfile file of an environment. You can add, modify, and delete ECS parameters, and then use the **apply** command to enable these settings.

ECS parameters are sorted into the **resources.ecs.autoScaling** category of the Wpfile file. The following table describes the ECS parameters.

Category	Parameter	Valid values	Default value	Description
resources.ecs.autoScaling	instanceNum	0 to 100	1	The number of required ECS instances.
	instanceType	A list of instance types in YAML format.	None	The list of instance types.
	securityGroupIds	A list of security group IDs in YAML format.	None	The list of security groups.
	keyPairName	The name of a key pair.	None	The name of the key pair.

Category	Parameter	Valid values	Default value	Description
	systemDiskSize	40 to 500	100	The size of the system disk.
	enableInternet	true and false	true	Specifies whether to enable an Internet IP address.

2.5 Server Load Balancer

In Web+, if multiple ECS instances exist in an environment, you can use Server Load Balancer (SLB) to distribute workloads across these ECS instances. With traffic distribution, SLB allows you to enhance service capabilities, remove single points of failure, and improve the availability of applications.

Background information

When you select a surrogate purchase for an SLB instance, Web+ helps you purchase the new SLB instance and configure listening ports and forwarding rules. Therefore, you can focus on your core business rather than using your time to learn about the underlying configurations of SLB. The default instance type for a surrogate SLB instance is **slb.s1.small**. The billing method is pay-as-you-go.

If you have created one or more SLB instances before using Web+, you can select the Import mode. This mode allows you to import these SLB instances and configure listening ports and server groups. After the configuration is complete, Web+ synchronizes ECS instances that reside in an environment to the specified server group.

In Web+, you can create an Internet-facing SLB instance to allow access to an application from the Internet. You can also create an internal SLB instance. This type of SLB instances only allows access to an application from applications on clients that reside in the same VPC as a client that hosts the application. The procedure to create an Internet-facing SLB instance or an internal SLB instance is similar. This topic takes the procedure to create an Internet-facing SLB instance as an example.

Surrogate SLB instances

1. On the Environment Details page, select Configurations, and view the **Internet-facing SLB** setting.

2. Turn on the **Enable Internet-facing SLB** switch.
3. Select **Surrogate Purchase** as the **Instance Source**.
4. Enter a port number in the **SLB Listening Port** setting.

The listening port of the SLB instance. You can use the port to access applications over the Internet. If an SLB instance is purchased by Web+ on your behalf, Web+ will help you create and maintain a listening port.

5. Configure the **SLB Protocol**.

The protocol type of request traffic that Server Load Balancer supports routing to different ports on instances. Available protocols include TCP and HTTP. If you specify HTTP as the SLB protocol, you can configure a forwarding policy. Only HTTP request traffic that conforms to the specified forwarding policy will be routed to instances that reside in the current environment.

6. Configure the **SLB Forwarding Policy**. The setting is only available when you select HTTP for the SLB Protocol setting.

A valid forwarding policy includes at least a fully qualified domain name (FQDN) or a directory. Examples of valid forwarding policies are listed as follows:

- www.taobao.com/test
- www.taobao.com
- /test

Self-imported SLB instances

1. On the Environment Details page, select Configurations, and view the **Internet SLB** setting.
2. Turn on the **Enable Internet SLB** switch.
3. Select **Import** as the **Instance Source**.
4. On the list of **SLB instances**, select an existing SLB instance.
5. Enter a port number in the **SLB Listening Port**.

The listening port of the SLB instance. You can use the port to access applications over the Internet. If you import existing SLB instances, the listening port is used as part of an Internet access address. You must ensure that the port is identical to the port that is linked to the specified VServer group.

6. Configure the **SLB Server Group Type**.

- **Default Server Group:** includes one or more ECS instances that accept requests from clients. If no VServer group is available, all requests are forwarded to ECS instances that reside in the default server group.
- **VServer Group:** You can select an existing VServer group or click **Create VServer Group** to create a new VServer group. Then, you can click the ID of a VServer group to go to the SLB console to configure listening ports and forwarding policies.

When you want to forward requests to different backend servers or forward requests based on domain names or URLs, you can specify a VServer group.

When you scale an environment, Web+ will synchronize ECS instances to the specified VServer group.

Disable SLB

After you disable SLB, Web+ uses the following policies to release SLB instances:

- **Surrogate SLB instances:** After you disable SLB, Web+ removes all ECS instances, forwarding policies, and listeners that reside in the specified VServer group of an environment. The SLB instance will also be released. If the SLB instance is being used by other applications, Web+ removes all ECS instances, forwarding policies, and listeners but retains the SLB instance.
- **Self-imported SLB instances:** Web+ removes all ECS instances that reside in the specified VServer group of an environment but retains the SLB instance.

Use the CLI tool to configure SLB

The CLI tool uses the Wpfile file to configure SLB. You can use the **wpctl dump** command to retrieve the Wpfile file of an environment. You can use the CLI tool to add, modify, and delete SLB settings. Then, you can use the **apply** command to enable the configuration.

In the Wpfile file, Internet-facing SLB settings are sorted into the **resources.slb.internet** category and internal SLB settings are sorted into the **resources.slb.intranet** category. An identical set of settings applies to both Internet-facing SLB and internal SLB. The following table describes the SLB settings.

Setting	Valid values	Default value	
enable	true and false	false	Specifies whether to enable SLB.

Setting	Valid values	Default value	
imported	true and false	false	Specifies whether to import an existing SLB instance.
importedGroupType	defaultGroup virtualGroup virtualGroup		The server group type for the imported SLB instance. Valid values: defaultGroup specifies the default server group and virtualGroup specifies a VServer group.
vServerGroupId	String		The ID of the VServer group. If you import an SLB instance and the SLB Server Group Type is set to VServer Group, you must specify a VServer group ID.
slbId	A valid SLB instance ID.	None	When you specify an SLB instance ID, Web+ configures the SLB instance based on your settings . Then, the SLB instance forwards specific traffic to ECS instances that reside in the current environment.
internetChargeType	<ul style="list-style-type: none"> • paybytraffic • paybybandwidth 	paybytraffic	The billing item . Valid values: paybytraffic specifies that you are billed by traffic volume and paybybandwidth specifies that you are billed by bandwidth.

Setting	Valid values	Default value	
loadBalancerSpec	<ul style="list-style-type: none"> slb.s1.small slb.s2.small slb.s2.medium slb.s3.small slb.s3.medium slb.s3.large 	slb.s1.small	The SLB instance type.
bandwidth	Integer	-1	The peak bandwidth parameter. When you set the billing method to paybybandwidth, the peak bandwidth parameter is available.
listenerPort	Integer	80	The listening port.
protocol	<ul style="list-style-type: none"> HTTP TCP 	HTTP	The listening protocol.
forwardingRule	A valid forwarding policy includes at least an FQDN or directory. Examples of valid forwarding policies are listed as follows: <code>www.taobao.com/test</code> , <code>www.taobao.com</code> , and <code>/test</code> .	None	The forwarding policy parameter. When you set the listening protocol to HTTP, the forwarding policy parameter is available.

2.6 ApsaraDB for RDS instances

Web App Service allows you to orchestrate RDS resources. You can add RDS instances to your environment by configuring the database type, database version, storage type, and other required settings.

Background information

Web App Service provides surrogate and self-imported ApsaraDB for RDS instances. To prevent data loss, Web App Service will not release the attached ApsaraDB for RDS instance

when you release an environment. You can log on to the [ApsaraDB for RDS console](#) to back up data and release these instances.

Web App Service uses [Key Management Service \(KMS\)](#) to create master keys to encrypt all database keys. Master keys created by Web App Service are free of charge. [DO NOT TRANSLATE] [DO NOT TRANSLATE]

ApsaraDB for RDS settings

Enable RDS

Turn on the **Enable RDS** switch to attach an RDS instance.

You can use the following method to attach an ApsaraDB for RDS instance to an environment:

- **Surrogate Purchase:** If you select this option, Web App Service will automatically purchase an ApsaraDB for RDS instance on your behalf after you configure a username, password, and other database-related settings. By default, Web App Service purchases a pay-as-you-go ApsaraDB for RDS instance on your behalf. After the process of a surrogate purchase is complete, you can change the billing method of the new instance to subscription in the [ApsaraDB for RDS console](#). For information about pricing for ApsaraDB for RDS, see [Pricing and billing items](#).
- **Import:** Before using Web App Service, you must first create ApsaraDB for RDS instances. Then, you can import an available instance to Web App Service. If you import an existing ApsaraDB for RDS instance, you do not need to configure the settings that are required when Web App Service purchases an ApsaraDB for RDS instance on your behalf. These settings include the instance type, database type and version, storage type, and storage capacity.

Disable RDS

Turn off the Enable RDS switch to detach an RDS instance from an environment.

Note: If you disable RDS, Web App Service will not automatically release the attached RDS instance to prevent data loss. You can back up data and release the instance in the [ApsaraDB for RDS console](#).

Database Type and Database Version

ApsaraDB for RDS supports a variety of database engines including MySQL, SQL Server, PostgreSQL, PPAS, and MariaDB TX. Several versions of each data engine are available. You

can select a database version based on your business requirements. For more information, see [Database engines](#).

Database Edition

Four database editions are available for RDS instances. For more information, see [#unique_17](#).

- **Basic:** This edition provides a single node. It separates computing from storage and provides high performance at ultra-low costs.
- **HA:** This edition adopts a high-availability architecture that consists of a primary node and a secondary node. It is applicable to more than 80% of scenarios.
- **Cluster:** This edition is only available for ApsaraDB RDS SQL Server 2017 Enterprise. This edition leverages the AlwaysOn technology and provides one primary node, one secondary node, and a maximum of seven read-only nodes. This architecture allows you to perform horizontal scaling of a cluster to enhance read capabilities. When you purchase an RDS instance of the SQL Server 2017 Enterprise version, the HA architecture is applied by default. This architecture only includes a primary node, secondary node, and no read-only node.
- **Finance:** This edition uses a three-node architecture that includes one primary node and two secondary nodes. Data is replicated between the primary and secondary nodes to ensure data consistency and financial-grade reliability.

Storage types and storage capacity

To meet requirements for different scenarios, ApsaraDB for RDS provides you with three storage types. For more information, see [#unique_18](#).

- **Local SSD (Recommended):** indicates SSDs that reside in the same node as the database engine. You can store data on a local SSD to reduce I/O latency.
- **SSD:** indicates elastic block storage devices that are based on the distributed storage architecture. You can store data on SSDs to implement the compute-storage separation.
- **ESSD:** indicates next-generation disks that provide ultra-high performance. Enhanced SSDs (ESSDs) are based on the new-generation distributed block storage architecture. With 25 Gigabit Ethernet and remote direct memory access (RDMA) technologies, this architecture provides several benefits. These benefits include a maximum of one million random input/output operations per second (IOPS) for a single disk and low latency for single-line communication.

You can configure the storage capacity after you select a storage type.

Zone and VSwitch

Each zone is a distinct geographic location that resides in a region. Each zone has isolated power supplies and networks. A VSwitch is attached to a zone and used to connect RDS instances that reside in the zone. Multiple zones that reside in the same region are connected through high-speed links. The zone where you create an RDS instance can be the same as the zone where you create an ECS instance that runs an application. You can also create an RDS instance and ECS instance in different zones because no distinct difference between zones that reside in the same region.

Type

RDS instance types include General and Dedicated. For more information, see [#unique_19](#).

- **General:** indicates RDS instances that are assigned exclusive memory and I/O resources. However, these RDS instances share CPU and storage resources with other RDS instances of the General type on the same physical host.
- **Dedicated:** indicates RDS instances that are assigned exclusive CPU, memory, storage, and I/O resources. These instances will not be affected by failures that occur on other instances that reside in the same physical host. The performance of these instances is persistent and stable. The **Dedicated host** type is the top level of the Dedicated type. Each RDS instance of this type occupies all the resources of a physical host.

Database Name

You need to specify a database name in the Web App Service console. For surrogate instances, Web App Service will automatically create a database if the specified database does not exist in the RDS instance. For self-imported instances, you must ensure that the specified database is available on the RDS instance.

Account Name and Password

You need to specify an account name and password for a database in Web App Service. For surrogate instances, Web App Service will automatically create an account for you if no account is available on an RDS instance. For self-imported instances, you must ensure that the specified account is available on the RDS instance and is granted the required permissions.

Use the CLI tool to configure ApsaraDB for RDS

The CLI tool uses the Wpfile file to configure ApsaraDB for RDS instances. You can use the `wpctl dump` command to retrieve the Wpfile file of a deployment environment. You can add

, modify, and remove RDS-related parameters. Then, you can use the apply command to apply the update.

RDS-related parameters are sorted into the **resources.rds** category of the Wpfile file. The following table provides a list of parameters and the description for each parameter.

Parameter	Valid value	Default value	Description
enable	true and false	false	Specifies whether to enable RDS.
imported	true and false	false	Specifies whether to use an existing RDS instance.
rdsId	Valid RDS instance IDs	N/A	The ID of an RDS instance. If you need to import an existing RDS instance, you must specify the ID of the instance.
zoneId	Valid zone IDs	N/A	The ID of the zone where the RDS instance resides. You can call the DescribeRegions operation to query the zone of the RDS instance.
instanceChargeType	PrePaid and PostPaid	PostPaid	The payment method for the RDS instance.
periodUnit	Month and Year	Month	The subscription duration unit of the RDS instance. This parameter is required if the billing method of the RDS instance is subscription.

Parameter	Valid value	Default value	Description
period	<ul style="list-style-type: none"> 1, 2, and 3 when the value of the periodUnit parameter is Year 1, 2, 3, 4, 5, 6, 7, 8, 9, 12, 24, and 36 when the value of the periodUnit parameter is Month 	1	The subscription duration of the RDS instance. This parameter is required if the billing method of the RDS instance is subscription.
VSwitches	Valid VSwitch IDs.	N/A	The ID of the VSwitch in the zone where the RDS instance resides.
engine	Valid database types, including MySQL, SQLServer, PostgreSQL, PPAS, and MariaDB.	N/A	The database type .
engineVersion	Valid database versions: <ul style="list-style-type: none"> MySQL: 5.5, 5.6, 5.7, and 8.0. SQL Server: 2008r2, 2012, 2012_ent_ha, 2012_std_ha, 2012_web, 2016_ent_ha, 2016_std_ha, 2016_web, and 2017_ent PostgreSQL: 9.4 and 10.0 PPAS: 9.3 and 10.0 MariaDB: 10.3 	N/A	The database version of the RDS instance.
storageType	Valid storage types: <ul style="list-style-type: none"> local_ssd/ephemeral_ssd: local SSD (recommended) cloud_ssd: SSD cloud_essd: ESSD 	N/A	The storage type of the RDS instance.
storageSize	Integers	100	The storage capacity of the RDS instance. Unit : GB.

Parameter	Valid value	Default value	Description
dbInstanceClass	Valid instance types	N/A	The type of the RDS instance . For more information about valid instance types, see the ApsaraDB for RDS documentation.
databaseName	Strings.	webplus	The database name of the RDS instance.
characterSetName	Valid character sets: <ul style="list-style-type: none"> MySQL or MariaDB: utf8, gbk, latin1, and utf8mb4. SQL Server: Chinese_PRC_CI_AS, Chinese_PRC_CS_AS, SQL_Latin1_General_CP1_CI_AS , SQL_Latin1_General_CP1_CS_AS, and Chinese_PRC_BIN. 	<ul style="list-style-type: none"> MySQL or MariaDB: utf8mb4. SQL Server: Chinese_PRC_CI_AS. 	The character set of the RDS instance.
accountName	Strings	webplus	The database account name of the RDS instance.
accountPassword	Strings.	N/A	The password corresponding to the database account name.
category	Valid database editions: <ul style="list-style-type: none"> Basic: the Basic edition HighAvailability: the HA edition AlwaysOn: the Cluster edition Finance: the Finance edition (only available on the China site). 	N/A	The database edition.

Attach an RDS instance to an application

After you enable RDS for an environment, Web App Service uses environment variables to pass the required information about the database connection to the application that run

in the environment. The application retrieves the required information about the database connection from the following environment variables:

- `WP_RDS_ENGINE`: the database type such as MySQL.
- `WP_RDS_CONNECTION_ADDRESS`: the connection string of the database, for example, `rm-xxxx.mysql.rds.aliyuncs.com`
- `WP_RDS_PORT`: the port number that is used to connect to the database, for example, 3306
- `WP_RDS_ACCOUNT_NAME`: the username that is used to access the database
- `WP_RDS_ACCOUNT_PASSWORD`: the password corresponding to the username
- `WP_RDS_DATABASE`: the name of the database

2.7 ApsaraDB for Redis instances

Web App Service allows you to orchestrate ApsaraDB for Redis resources. To add a Redis instance to your environment, you must configure the database type, database version, storage type, and other required settings for the instance.

Background information

ApsaraDB for Redis is a database service that is compatible with open-source Redis protocols and supports hybrid storage with a blend of memory and hard disks. With reliable dual-node hot standby and scalable clusters, Redis meets your requirements for high-performance read/write operations and flexible configuration changes.

After you enable Redis, the following methods are available for you to add an ApsaraDB for RDS instance to an environment:

- **Surrogate Purchase:** If you select this option, Web App Service will automatically purchase a Redis instance on your behalf. You must first specify the billing method, version type, architecture type, number of shards, node type, instance type, and password for the instance. A pay-as-you-go billing method is for short-term use. The pay-as-you-go billing method allows you to pay only for the number of resources that you use. Compared with a pay-as-you-go billing method, a subscription billing method is more cost-effective for long-term use. The subscription billing method allows you to pay for a service based on the subscription duration. For more information about Redis pricing, see [#unique_21](#).
- **Import:** If you select this option, you must create ApsaraDB for RDS instances before using Web App Service. Then, you can import an existing instance to Web App Service.

If you import an existing Redis instance, you do not need to specify the billing method, version type, architecture type, number of shards, node type, and other parameters for the instance.

You can use Web App Service to purchase Redis instances on your behalf or imported Redis instances. To ensure data security, Redis instances remain unchanged even if the environment that hosts these instances is released. You can log on to the [Redis console](#) to back up data and release instances.



Precautions


- You cannot switch between billing methods. The billing methods include pay-as-you-go and subscription.
- You can only change the password for each subscription instance. Changes to other parameters are invalid.
- You can change the password and instance type for each pay-as-you-go instance. Changes to other parameters are invalid.
- When you add pay-as-you-go or subscription Redis instances of the surrogate purchase type, a default account is created for each Redis instance. You can leave the password field blank.

Redis instances of the surrogate purchase type

1. On the Environment Details page, click **ApsaraDB for Redis**.
2. On the **ApsaraDB for Redis** page, turn on the **Enable Redis** switch.
3. Select **Surrogate Purchase** in the **Instance Source** field.
4. Select **Billing Method**.
 - **Pay-As-You-Go**: You pay for the number of resources after you consume the resources. When you release or scale down an environment, instances that are attached to the instance will not be released. A pay-as-you-go billing method is for short-term use.
 - **Subscription**: You need to set a subscription duration and pay for an instance upfront based on the subscription duration. When you release or scale down an environment, instances that are attached to the environment will not be released. Compared to a pay-as-you-go billing method, a subscription billing method is more cost-effective for long-term use.

5. Select **VSwitch** from the drop-down list. A VSwitch is a basic building block to create a VPC. If no VSwitch is available in a VPC, you must [#unique_22](#).
6. Set the required parameters for a **Redis instance**.

Parameter	Description
Version	<ul style="list-style-type: none"> • Community: supports hybrid storage that integrates memory with hard disks and open-source Redis protocols . • Enterprise: The Enterprise version and the Community version complement each other in areas, such as performance, storage media, and data structure. .
Database Edition	<p>Available database editions for Redis instances of the Enterprise version include:</p> <ul style="list-style-type: none"> • Enhanced Performance. For more information, see #unique_23. • Hybrid Storage. For more information, see #unique_24. <div style="background-color: #f2f2f2; padding: 5px;">  Note: Database editions are not available for Redis instances of the Community version. </div>
Database Version	<p>Available database versions for Redis instances include:</p> <ul style="list-style-type: none"> • 2.8 • 4.0 • 5.0 <div style="background-color: #f2f2f2; padding: 5px;">  Note: <ul style="list-style-type: none"> • If the required database version does not appear, we recommend that you specify a different database edition or instance version. • Redis instances of the 2.8 database version are about to be recalled. To obtain more features and higher performance, we recommend that you create Redis instances of the latest database version. </div>
Architecture Type	<ul style="list-style-type: none"> • Cluster • Standard • Read/Write Splitting <p>For more information, see #unique_25.</p>

Parameter	Description
Shards	The number of shards for a Redis instance of the Cluster architecture type.
Node Type	Only the Dual-copy node type is available for Standard and Cluster Redis instances. To ensure persistent data storage, Redis instances of the Dual-copy node type adopt hot standby. The architecture includes a combination of a primary node and a secondary node for real-time failover.
Instance Type	<p>Each instance type includes a set of configurations, such as the memory size, maximum number of connections, and bandwidth limit. For more information, see #unique_26.</p> <div style="background-color: #f0f0f0; padding: 10px;"> <p> Note: After you create a Redis instance, metadata for a database is generated and uses a minimum amount of storage space.</p> <ul style="list-style-type: none"> • On a Redis instance of the Standard architecture type, the size of metadata ranges from 30 MB to 50 MB. • On each shard of a Redis instance of the Cluster architecture type, the size of metadata ranges from 30 MB to 50 MB. In a cluster, the total amount of storage space that metadata uses is the total size of metadata on each shard. </div>

7. Enter a **Password**.
8. In the upper-right corner of the page, click **Edit Configuration** to apply the update.

Import an SLB instance

1. On the **ApsaraDB for Redis** page, turn on the **Enable Redis** switch.
2. **Instance SourceSelect Import.**
3. In the **Redis Instance** list, select an existing Redis instance.
4. Enter a valid password for the default account of the Redis instance.
5. In the upper-right corner of the page, click **Edit Configuration** to apply the update.

Use the CLI tool to configure the Redis instance

The CLI tool uses the Wpfile file to configure Redis instance. You can use the `wpctl dump` to retrieve the Wpfile file of a deployment environment. You can add, modify, and remove Redis-related parameters. Then, you can use the `apply` command to apply the update.

The **resources.redis** section of the Wpfile file contains Redis-related parameters. The following table describes the parameters.

Parameter	Name	Valid value	Default value	Description
enable	[DO NOT TRANSLATE]	true and false	false	Specified whether to enable Redis.
imported	[DO NOT TRANSLATE]	true and false	false	Specified whether to import an existing Redis instance.
redisid	[DO NOT TRANSLATE]	Valid Redis instance IDs	N/A	The ID of the Redis instance . If you want to import an existing Redis instance, you must specify the parameter.
engineVersion	[DO NOT TRANSLATE]	2.8, 4.0, and 5.0	5.0	The database version for the Redis instance.
instanceChargeType	[DO NOT TRANSLATE]	PrePaid and PostPaid	PostPaid	The billing method for the surrogate Redis instance.
periodUnit	[DO NOT TRANSLATE]	Year and Month	Month	The unit of the subscription duration.

Parameter	Name	Valid value	Default value	Description
period	[DO NOT TRANSLATE]	If you specify Year for the periodUnit parameter, the valid values of the parameter include 1, 2, and 3. If you specify Month for the periodUnit parameter, the valid values of the parameter include 1, 2, 3, 4, 5, 6, 7, 8, 9, 12, 24, and 36.	1	The subscription duration.
password	[DO NOT TRANSLATE]	letters, digits, and special characters Special characters include @#\$%^&* () _+-.=.	N/A	The password for the Redis account. The password must be 8 to 32 characters in length.
instanceClass	[DO NOT TRANSLATE]	Valid instance types	N/A	The instance type of the Redis instance.
vSwitchId	[DO NOT TRANSLATE]	Valid VSwitch IDs	N/A	The ID of the VSwitch for the zone where the Redis instance resides.
zoneId	[DO NOT TRANSLATE]	Valid zone IDs	N/A	The ID of the zone where the Redis instance resides.

Attach a Redis instance to an application

After you enable Redis for an environment, Web App Service uses environment variables to pass the required information about database connection to the application that runs

in the environment. The application retrieves the required information about database connection from the following environment variables:

- `WP_REDIS_CONNECTION_ADDRESS`: includes the internal endpoint of a Redis instance.
- `WP_REDIS_PORT`: includes a port number that is used to connect to a Redis instance.
- `WP_REDIS_ACCOUNT_NAME`: includes the name of an account that is used to connect to a Redis instance.
- `WP_REDIS_ACCOUNT_PASSWORD`: includes a password that corresponds to the account.
- `WP_REDIS_ENGINE_VERSION`: includes the version of a Redis instance.

2.8 Configure a service port

A service port is opened by the backend process of an application on an instance. After you configure a service port, Web+ opens connections on this service port for an application that provides services to clients.

Procedure

1. Log on to the [Web+ console](#).
2. On the **Overview** page, click **View All** in the upper-right corner of the **Last Updated Environments** section.
3. On the **Applications and Environments** page, click the > icon next to the name of an application to view a list of related environments.



Note:

On the **Environments** list, the most recently updated four environments are displayed. If the required environments is displayed on the list, you can click the name to go to the Environment Details page.

4. Click the name of an Environment to open the **Environment Details** page.
5. In the **Applications** section, click **Basic Information**, and enter a port number in the Service Port setting.
6. After the configuration is complete, click **Change Configuration** in the upper-right corner to apply the update.

2.9 Configure health checks

Web+ provides you with many different types of health checks. You can use the feature to check the health status of an application. After you create health checks, Web+ performs

specific actions on an application after the health status is retrieved. For example, if a health check for an ECS instance fails, Web+ restarts the ECS instance. If the health status of an application instance is abnormal, Web+ restarts the instance.

Introduction to health checks

In Web+, health checks are categorized into two types:

- **Vitality checks for application instances:** checks whether the status of applications is healthy. If a health check for an application instance fails, Web+ restarts the instance. If an instance passes a health check, no action is performed.
- **Response checks for applications:** checks whether the status of applications is ready. This type of health check is used to check whether instances are ready to handle client requests. Many applications may require an extended period of time to start. For example, some applications may need to load large amounts of data from disks or applications may have to wait until an external dependent module is started. In such cases, applications cannot provide services even if the related processes are running at the backend. In this scenario, you can use health checks to monitor the health status of applications.

Health checks are separated into Layer-4 checks that are provided by the TCP protocol and Layer-7 checks that are provided by the HTTP protocol.

- A TCP health check sends SYN handshake packets to an instance to check whether the status of the instance is healthy.
- An HTTP health check simulates the process of using a Web browser to access resources, which sends HEAD or GET requests to an instance to check whether the instance is healthy. If the value of a status code returned from a health check URL ranges from 200 to 400, it indicates a successful health check.

If your businesses are highly sensitive to traffic load, frequent health checks may impact the availability of normal services. You can use multiple methods to mitigate any impact on your businesses based on actual requirements. These methods include reducing the health check frequency, increasing the health check interval, and changing Layer-7 health checks to Layer-4 health checks. We recommend that you do not disable health checks because it may affect business continuity.

Create or update health checks

1. Log on to the [Web+ console](#).

2. On the **Overview** page, click **View All** in the upper-right corner of the **Last Updated Environments** section.
3. On the **Applications and Environments** page, click the > icon next to the name of an application to show a list of related environments.



Note:

On the **Environments** list, the most recently updated four environments are displayed. If the required environment is displayed on the list, you can click the name of the environment to go to the Environment Details page.

4. Click the name of an environment to go to the Overview tab of the **Environment Details** page.
5. In the left-side navigation pane, click **Configurations** to go to the Configurations page.
6. In the **Applications** section, click **Health Check** to configure or change the required health check settings.

Setting	Description
Health Check Type	The protocol type for health checks. Available protocol types include TCP and HTTP. <ul style="list-style-type: none"> • TCP health checks are configured to send handshake packets to instances over an IP network and open connections on a port to check whether each instance is healthy. • HTTP health checks are configured to send HEAD or GET requests to the specified health check URL of each instance to check whether each instance is healthy.
Health Check URL	The format of a health check URL is <code>http://<an IP address>:<a port number>/<a directory></code> . The URL to which you send health check requests. You can check whether an instance is healthy based on a value returned from the specified URL. A returned value that ranges from 200 to 400 indicates a successful health check.
Unhealthy Threshold	If the number of health check retries exceeds the specified threshold, it indicates a failed health check.
Interval (in seconds)	The amount of time between a failed health check and the next health check.

Setting	Description
Timeout (in seconds)	The amount of time to wait for a health check response. If no health check response is returned after the timeout period ends, it indicates a failed health check.

7. After the configuration is complete, click **Change Configuration** to enable the update.

Disable health checks

If you want to disable health checks for an environment, you can only disable HTTP health checks. You cannot disable TCP health checks.

1. View the Health Check section. If you select **HTTP** for the **Health Check Type** setting, you can enter a health check URL in the **Health Check URL** setting.
2. Click **Change Configuration** in the upper-right corner to apply the update.

2.10 Commands and lifecycle hooks

Web+ allows you to manage environments based on your business needs by customizing start commands, stop commands, and lifecycle hooks.

Configure commands or lifecycle hooks on the Configurations page

You can configure commands or lifecycle hooks when creating a new environment and when changing the configurations of a running environment.

Configure commands or lifecycle hooks when creating a new application and environment

1. Log on to the [Web+ console](#).
2. On the **Overview** page, click **Create** in the upper-right corner of the **Last Updated Environments** section. Follow the instructions to configure the required settings in the **Basic Information** and **Environment Information** steps. Then, configure the required settings in the **Configurations** step.
3. In the **Configurations** step, select **Custom** for the **Pre-defined Configuration** setting.
4. On the list of displayed settings, click **Commands** or **Lifecycle Hooks** in the Applications section and configure the required settings.
5. Click **Creation Complete** to create a new environment.

Configure commands or lifecycle hooks when updating an Environment environment

1. Log on to the [Web+ console](#).

2. On the **Overview** page, click **View All** in the upper-right corner of the **Last Updated Environments** section.
3. On the **Applications and Environments** page, click the ID of an application for which you want to create a new environment.
4. On the **Application Details** page, click **Create Environment** in the upper-right corner, configure the required settings in the Environment Information step, and then go to the **Configuration** step.
5. In the **Configurations** step, select **Custom** for the **Pre-defined Configuration** setting.
6. On the list of Environments
 1. section.
 2. On the **Applications and Environments** page, click the > icon next to the name of an application to show a list of linked environments.
 3. Click the name of an environment to go to the Overview tab of the **Environment Details** page.
 4. In the left-side navigation pane, click **Configurations** to go to the Configurations page.
 5. On the Configurations page, click **Commands** or **Lifecycle Hooks** in the Applications section and configure the required settings.
 6. Click **Change Configuration** to update the environment.

Commands

In most cases, Web+ predefines default stop and start commands for each type of technology stacks. Web+ uses systemd to manage user processes by default.



Note:

- A start command must run as a frontend application. When the start command is complete, the backend service that corresponds to the command is also stopped.
- We recommend that you always include the **systemctl stop wpweb** snippet in a stop command to ensure that a service can be terminated as expected. You can add custom statements before and after the snippet to complement a stop command.
- The working directory of a start command is the same as the local directory that stores the deployment package of an application to which the start command applies.
- You must use the admin credential to run a start command.
- If a deployment package includes Procfile, the specified start command that applies to an application built by the deployment package will not take effect.

Lifecycle hooks

When Web+ performs update actions on instances, specific lifecycle hooks will be triggered. You can specify custom commands or scripts for these lifecycle hooks to perform specific actions. These actions include modifying configuration files, registering or deregistering services, sending notifications, and downloading files.

- **PostPrepareEnv:** After a environment is initialized, the lifecycle hook is executed. The lifecycle hook only runs once for each instance. You cannot read environment variables while the lifecycle hook is running.
- **PreInstallStack:** Before you install or change fundamental software that is included in a technology stack, the lifecycle hook is executed. You cannot read environment variables while the lifecycle hook is running.
- **PostInstallStack:** After you install or change fundamental software that is included in a technology stack, the lifecycle hook is executed. You cannot read environment variables while the lifecycle hook is running.
- **PrePrepareApp:** Before an update occurs on an application such as a change in deployment package version, the lifecycle hook is executed. You cannot read environment variables while the lifecycle hook is running.
- **PostPrepareApp:** Before an update occurs on an application such as a change in deployment package version, the lifecycle hook is executed. You can use environment variables while the lifecycle hook is running.
- **PreStart:** Before an application is started, the lifecycle hook is executed. You can use environment variables while the lifecycle hook is running.
- **PostStart:** After an application is started, the lifecycle hook is executed. You can use environment variables while the lifecycle hook is running.
- **PreStop:** Before an application is stopped, the lifecycle hook is executed. You can use environment variables when the lifecycle hook is running.
- **PostStop:** After an application is stopped, the lifecycle hook is executed. You can use environment variables while the lifecycle hook is running.



Note:

- You must use **root** permissions to execute lifecycle hooks.
- A failed execution of a lifecycle hook results in a failed update and the termination of the update. You can avoid update errors regardless of whether the execution result of a lifecycle hook is successful or fails. We recommend that you add the **;exit 0** snippet at

the end of a lifecycle hook command to ensure the successful execution of a lifecycle hook.

- You can use environment variables in some types of lifecycle hooks. For more information about available environment variables, see [Environment variables](#).
- A lifecycle hook must be completed within two minutes. Otherwise, an update fails and is terminated. If a task requires more than two minutes to complete, we recommend that you run a script at the backend to complete the task.
- The working directory of lifecycle hooks is `/opt/webplus/data/envdata/wproot/`. You can use the `cd` command to move to the specified directory.
- We recommend that you specify the absolute path of an executable file that points to a command or script in a lifecycle hook.

2.11 Environment variables

You can specify a set of environment variables specific to each application. You can use these environment variables to change settings for an application at any time after you deploy the application. Environment variables that are specified for an application cannot be applied to a different application. You can specify the same environment variable for multiple applications but the value of the environment variable is unique to each application.

Configure environment variables

1. On the Configurations tab of the Environment Details page, view the **Start Configuration** field.
2. Enter a **Variable Name** and **Variable Value**. If you want to add more environment variables, click **Add**.



Notice:

[Predefined environment variables](#) have a higher priority than console settings. Do not repeatedly configure environment variables. This prevents console settings from being overridden.

3. After the configuration is complete, click **Change Configuration** in the upper-right corner of the Configurations page to enable the configurations.

Predefined environment variables

In addition to custom environment variables, Web+ provides with you predefined environment variables. You can configure these environment variables based on your requirements. The following table lists these environment variables, sample values, and the description of each environment variable.

Variable name	Example	Description
WP_ENV_ID	we-5d3e9e9a4cb0d77689e9****	The ID of the environment.
WP_ENV_NAME	test_env	The name of the environment.
WP_ENV_CREATE_USER	test.user	The name of the user who creates the environment.
WP_ENV_CREATE_TIME	1564384923001	The creation time for the environment.
WP_CHANGE_ID	wc-5d3f096e4cb0d77689e9****	The update ID of the latest environment change.
WP_CHANGE_TRIGGER_FROM	Console	The source from which the latest environment change initiates.
WP_CHANGE_TRIGGER_USER	test.user	The user who initiates the latest environment change.
WP_CHANGE_NAME	apply.env	The name of the update for the latest environment change.
WP_CHANGE_CREATE_TIME	1564412270849	The start time of the latest environment change.
WP_CHANGE_REQUEST_ID	911B19AF-DB53-450D-84FB-2A2307DC****	The request ID for the latest environment change.
WP_APP_ID	wa-5d3e9e994cb0d77689e9****	The ID of the application.
WP_APP_NAME	test_app	The application name.
WP_APP_CATEGORY	PHP	The technology stack type of the application.
WP_APP_CREATE_TIME	1564384921921	The creation time of the application.

Variable name	Example	Description
WP_APP_CREATE_USER	test.user	The name of the user who creates the application.
WP_PKG_VERSION_ID	wp-5d3e9e9a4cb0d77689e9****	The version ID of the deployment package.
WP_PKG_VERSION_LABEL	20190729.152153	The version of the deployment package.
WP_PKG_VERSION_ETAG	266FB8E532E1D924BDAF C65A261E****	The ETag value of the deployment package version . The value is produced by using the MD5 message-digest algorithm.
WP_PKG_VERSION_SOURCE	resources/Php1/versions/ 20190729.152153/webplus- demo-php.zip	The storage path of the deployment package for the version.
WP_PKG_VERSION_CREATE_TIME	1564384922317	The creation time of the deployment package version .
WP_PKG_VERSION_CREATE_USER	test.user	The user who creates the deployment package version .
WP_INSTANCES_ID	i-2ze2u0df969vuwow****	The ID of the ECS instance.
WP_INSTANCES_PRIVATE_IP	172.17.82.37	The private IP address of the ECS instance.
WP_RDS_ENGINE	MySQL	The database engine for the ApsaraDB for RDS instance.
WP_RDS_CONNECTION_ADDRESS	rm-2ze49r7276jbi****.mysql. rds.aliyuncs.com	The internal endpoint of the ApsaraDB for RDS instance.
WP_RDS_PORT	3306	The port number of the ApsaraDB for RDS instance.
WP_RDS_ACCOUNT_NAME	webplus	The account name of the ApsaraDB for RDS instance.
WP_RDS_ACCOUNT_PASSWORD	*****	The password of the ApsaraDB for RDS instance.
WP_RDS_DATABASE	webplus	The name of the ApsaraDB for RDS instance.

Variable name	Example	Description
APP_HOME	/home/admin/app/testapp	The local folder of the deployment package.
TOMCAT_HOME	/home/admin/tomcat8	The installation folder of Tomcat.
WP_SERVICE_PORT	8080	The service port.

**Notice:**

- When you log on to an ECS instance by using Secure Shell (SSH), environment variables that are specified in Web+ will not be loaded.
- Web+ cannot automatically configure a service port for applications of the following technology stack types: Java, Go, Node.js, and Native. You can change the value of the `$WP_SERVICE_PORT` environment variable to configure a service port to avoid health check failures.

2.12 Configure parameters for a Java virtual machine

Java virtual machine (JVM) parameters are used to configure container parameters when an application starts. You can configure JVM parameters to help reduce the overhead of garbage collection (GC), reduce server response time, and improve overall throughput. If you do not set container parameters, JVM parameters are allocated by default.

Introduction to JVM

JVM is a virtual machine that enables hosts to run Java applications or applications that are compiled to Java bytecode based on heaps. Sun Microsystems is the first company to introduce and implement JVM. As part of a Java platform, JVM is used to run Java applications.

JVM has a complete hardware architecture, such as processors, heaps, and registers, and a set of corresponding instructions. JVM allows Java applications to run on any platforms without any changes by implementing underlying communications with a specific operating system. However, these Java applications must be compiled to target code or bytecode that is compatible with JVM. JVM allows Java bytecode that is compiled from applets or applications to run on a specific CPU by converting Java bytecode into machine code specific to the CPU.

Set JVM parameters

1. Log on to the [Web+ console](#).
2. On the **Overview** page, click **View All** in the upper-right corner of the **Last Updated Deployment Environment** section.
3. On the **Applications and Deployment Environments**, click > on the left side of an application to view a list of deployment environments related to the application.



Note:

On the **Deployment Environments** list, the most recently updated four deployment environments are displayed. If the required deployment environment is displayed on the list, you can click the specific name to go to the Deployment Environment Details page.

4. Click the specific name of a deployment environment to open the **Overview** page of the deployment environment.
5. In the left-side navigation pane, click **Configurations** to open the Configurations page.
6. In the **Applications** section, click **JVM** to view settings and specify JVM parameters.
7. After the configuration is complete, click **Edit Configuration** in the upper-right corner to apply the configuration.

3 Configure HTTPS for a deployment environment

In Web App Service, you can enable HTTPS for a deployment environment. This feature helps you enhance the security of a website. You do not need to enable your application to support HTTPS. Web App Service removes Secure Sockets Layer (SSL) certificates from incoming requests after you enable HTTPS for a deployment environment in Web App Service. Then, Web App Service relays these requests to a backend application by using HTTP. In Web App Service, you can enable HTTPS on a reverse proxy or a Server Load Balancer (SLB) instance.

(Optional) Obtain a certificate

You can use Web App Service self-signed certificates. This type of certificate is issued by a non-trusted certification authority (CA). When being visited with self-signed certificates, a website may be considered invalid. We recommend that you use this type of certificate only for testing.

You can also obtain certificates from a trusted CA and upload these certificates to Web App Service.

- **Alibaba Cloud SSL Certificates:** You can purchase or obtain certificates at no extra charge in the [SSL Certificates](#) console.
- **Third-party CA:** For information about how to obtain certificates from a third-party CA, see the documentation that is provided by the CA.

Configure HTTPS on a reverse proxy

If you enable a reverse proxy in your environment, you can configure HTTPS on the proxy. If HTTPS is enabled, the reverse proxy listens on port 443. Otherwise, the reverse proxy listens on port 80. After you enable HTTPS on the reverse proxy, the reverse proxy converts incoming HTTPS requests into HTTP request by removing SSL certificates. Then, the reverse proxy forwards HTTP requests to the backend application.



Notice:

You cannot configure HTTPS in the Reverse Proxy section. This is only applicable if you try to configure HTTP or HTTPS as a protocol type in the Internet Load Balancer SLB or Intranet Load Balancer SLB section.

1. Log on to the [Web+ console](#).
2. On the **Overview** page, click **View All** in the upper-right corner of the **Last Updated Deployment Environment** section.
3. On the **Applications and Environments** page, click > on the left side of an application to view a list of deployment environments related to the application.
4. Click the name of an environment to go to the **Environment Details** page.
5. In the left-side navigation pane, click **Configurations**.
6. In the **Environment** section, click **Reverse Proxy**, turn on the **Enable Reverse Proxy** switch, select **Nginx (1.14.2)** in the Reverse Proxy Type field, and select **HTTPS** in the **Protocol** field.
7. Use one of the following methods to configure certificates:
 - **Auto-configure certificates:** Turn on the **Auto-configure Certificate** switch. After this switch is turned on, Web App Service generates self-signed server certificates for a reverse proxy. This type of certificate is issued by a non-trusted CA. When being visited with self-signed certificates, a website may be considered invalid. We recommend that you use this type of certificate only for testing.
 - **Upload certificates:** Turn off the **Auto-configure Certificate** switch if you want to use trusted CA certificates in a production environment. Then, obtain certificates from a trusted CA and upload these certificates to Web App Service. You need to upload a **public key certificate file** and a **private key file** to Web App Service for each certificate. You must keep private key files confidential to prevent against information leakage.
8. After the configurations are complete, click **Change Configuration** in the upper-right corner to apply the update.

Configure HTTPS on an SLB instance

If you want to use HTTPS in Web App Service and your environment includes a Server Load Balancer (SLB) instance, we recommend that you enable HTTPS on the SLB instance.

1. On the Environment Details page, select Configurations, and click **Internet Load Balancer SLB**.
2. Turn on the **Enable Internet SLB** switch.
3. Select **Surrogate Purchase** in the **Instance Source** field.

4. Enter 443 in the Listening Port field.

The SLB instance listens on port 443. You can use the port to access applications over the Internet. If an SLB instance is purchased by Web App Service on your behalf, Web App Service will help you create and maintain the listening port.

5. Select HTTPS in the SLB Protocol field.**6. Select a certificate in the Server Certificate section.**

You must add a certificate file to SLB before you enable HTTPS on an SLB instance. You can click one of the links in the Server Certificate section to manage or purchase certificates.

7. Configure SLB forwarding rules.

The common format of a forwarding policy is a combination of a fully qualified domain name (FQDN) and path. A valid forwarding policy must include either an FQDN or a path. The following lists examples of valid forwarding policies:

- www.taobao.com/test
- www.taobao.com
- /test

8. Click Change Configuration in the upper-right corner to apply the update.**Notice:**

- You cannot configure HTTPS on an SLB instance if you have enabled HTTPS on a reverse proxy.
- The method that is used to configure HTTPS on an Internet-facing SLB instance is the same as the method for an internal SLB instance. You can configure HTTPS on both an Internet-facing SLB instance and an internal SLB instance.

4 View metrics for a environment

In the Web+ console, metrics are displayed in separate diagrams on the Monitoring page and change over time. These metrics include CPU usage, memory usage, disk usage, read/write BPS for system disks, read/write IOPS for system disks, Internet bandwidth, and intranet bandwidth. You can click a diagram to view the details of a metric at a specific time.

Access the Monitoring page

1. Log on to the [Web+ console](#).
2. On the **Overview** page, click **View All** in the upper-right corner of the **Last Updated Environments** section.
3. On the **Applications and Environments** page, click > on the left side of an application to view a list of environments related to the application.



Note:

On the **Environments** list, the most recently updated four environments are displayed. If the required environment is displayed on the list, you can click the specific name to go to the Environment Details page.

4. Click the specific name of an environment to open the **Overview** page of the environment.
5. In the left-side navigation pane, click **Monitoring** to view the metrics of the environment.

View monitoring data that is generated in a specific time range

You can select a time range in the upper-right corner of the Monitoring page. You can click the provided options to show the monitoring data that is generated the last **5 Minutes**, **1 Hour**, **6 Hours**, or **24 Hours**. You can also click **Custom** to search for up to 7 consecutive days of monitoring data within the last 31 days.

5 View events

The Events page provides you with a list of all events that are related to an environment and application versions included in the environment. An event is generated in Web+ when you create an environment or modify the resources of the environment.

Access the Events page

1. Log on to the [Web+ console](#).
2. On the **Overview** page, click **View All** in the upper-right corner of the **Last Updated Environments** section.
3. On the **Applications and Environments** page, click > next to an application to view a list of environments related to the application.



Note:

On the **Environments** list, the most recently updated four environments are displayed. If the required environment is displayed on the list, you can click the specific name to go to the Environment Details page.

4. Click the name of an environment to go to the **Overview** page of the environment.
5. In the left-side navigation pane, click **Events**.

On the **Events** page, all events related to the environment and application versions are displayed in chronological order by default.

Level

Events are categorized into the following types based on their level of severity: **INFO**, **ERROR**, and **WARN**.

Details

This column includes a detailed description and status for an operation of each event.

View events that are generated in a specific time range

You can select a time range in the upper-right corner of the **Events** page. You can click the provided options to show a list of events that are generated in the last **5 Minutes**, **1 Hour**, **6 Hours**, or **24 Hours**. You can also click **Custom** to customize a time range.

6 View logs

In Web+, operations logs are generated for ECS instances that are included in a environment. You can view these logs to troubleshoot issues that occur in applications or configuration files. You can search for logs of an ECS instance within a collection of operations logs. When you attempt to collect logs of an ECS instance, Web+ sends a request to the ECS instance. After retrieving the requested logs, Web+ uploads these logs to OSS. Then, you can download the logs in the Web+ console.

Log path on an ECS instance

Logs are stored in the default paths of an ECS instance that resides in your environment. Log paths for each application and Web server are listed as follows:

Log paths for Web+ agents on a Linux ECS instance

- Log path for the agent daemon process on an ECS instance: `/var/log/webplus/webplus-daemon.log`
- Log path for Web+ deployment logs: `/var/log/webplus/webplus-deployment.log`

Log paths for each application and Web server on a Linux ECS instance

- Log path for Java applications: `/home/admin/app/webplus-app.log`
- Log path for Tomcat applications: `/home/admin/tomcat{8}/logs`
- Log path for Nginx logs: `/var/log/nginx`
- Log path for Apache logs: `/var/log/httpd`

Supported environment variables where you can specify log paths

- `$HOME`: the user directory
- `$APP_HOME`: the root directory of applications.
- `$TOMCAT_HOME`: the root directory of Tomcat.

Collect logs

Before requesting logs on Web+, you must select an ECS instance, configure the log path, and then initiate a log request.


1. Log on to the [Web+ console](#).
2. On the **Overview** page, click **View All** in the upper-right corner of the **Last Updated Environments** section.




3. On the **Applications and Environments** page, click the > icon next to the name of an application to show a list of associated environments.

**Note:**

On the **Environments** list, the most recently updated four environments are displayed. If the required environment is displayed on the list, you can click the name to open the Environment Details page.

4. Click the name of a environment to open the **Environment Details** page.
5. In the left-side navigation pane, click **Logs**.
6. In the upper-right corner of the Logs page, select an instance from which you want to retrieve logs.
7. When you collect logs for the first time, click **Collect Log**. Then, click the Edit icon next to Log Path. In the **Configure Log Path** dialog box, enter a log path. If you want to collect multiple types of logs, click **Add** and repeat the action to add multiple log paths. After the configuration is complete, click **OK**.

After you configure log paths, the specified log paths are displayed in the upper-left corner of the page. You can click  to modify or delete one or more log paths.




8. On the Logs page, click **Collect Log** in the upper-right corner of the page. A record is displayed for a successful log request.
 - When you click  in the **Actions** column of a record, you will be directed to OSS where a log package resides.
 - You can click  in the **Actions** column of a record to delete the record.
 - You can click the > icon next to a record ID to show diagnostic information for each instance. Then, you can click  in the **Actions** column of a instance to download a log file for later analysis. You can also click an instance ID to go to the ECS console to manage the instance.

Collect diagnostic information

In Web+, you must request diagnostic information based on ECS instances.

1. In the upper-right corner of the **Logs** page, select an instance from which you want to retrieve diagnostic information.
2. On the left side of the **Logs** page, click the drop-down icon next to **Collect Log**, and select **Collect Diagnostic Information** from the drop-down list.

3. A record is displayed for a successful diagnostic information request.

- When you click  in the **Actions** column of a record, you will be directed to OSS where a package for diagnostic information resides.
- You can click  in the **Actions** column of a record to delete the record.
- You can click the > icon next to a record ID to show diagnostic information for each instance. Then, you can click  in the **Actions** column of a instance to download a log file for later analysis. You can also click an instance ID to open the ECS console to manage the instance.