阿里云
日志服务

用户指南

文档版本：20180918

法律声明

阿里云提醒您在阅读或使用本文档之前仔细阅读、充分理解本法律声明各条款的内容。如果您阅读或使用本文档，您的阅读或使用行为将被视为对本声明全部内容的认可。

1. 您应当通过阿里云网站或阿里云提供的其他授权通道下载、获取本文档，且仅能用于自身的合法合规的业务活动。本文档的内容视为阿里云的保密信息，您应当严格遵守保密义务；未经阿里云事先书面同意，您不得向任何第三方披露本手册内容或提供给任何第三方使用。
2. 未经阿里云事先书面许可，任何单位、公司或个人不得擅自摘抄、翻译、复制本文档内容的部分或全部，不得以任何方式或途径进行传播和宣传。
3. 由于产品版本升级、调整或其他原因，本文档内容有可能变更。阿里云保留在没有任何通知或者提示下对本文档的内容进行修改的权利，并在阿里云授权通道中不时发布更新后的用户文档。您应当实时关注用户文档的版本变更并通过阿里云授权渠道下载、获取最新版的用户文档。
4. 本文档仅作为用户使用阿里云产品及服务的参考性指引，阿里云以产品及服务的”现状“、“有缺陷”和“当前功能”的状态提供本文档。阿里云在现有技术的基础上尽最大努力提供相应的介绍及操作指引，但阿里云在此明确声明对本文档内容的准确性、完整性、适用性、可靠性等不作任何明示或暗示的保证。任何单位、公司或个人因为下载、使用或信赖本文档而发生任何差错或经济损失的，阿里云不承担任何法律责任。在任何情况下，阿里云均不对任何间接性、后果性、惩戒性、偶然性、特殊性或刑罚性的损害，包括用户使用或信赖本文档而遭受的利润损失，承担责任（即使阿里云已被告知该等损失的可能性）。
5. 阿里云网站上所有内容，包括但不限于著作、产品、图片、档案、资讯、资料、网站架构、网站画面的安排、网页设计，均由阿里云和/或其关联公司依法拥有其知识产权，包括但不限于商标权、专利权、著作权、商业秘密等。非经阿里云和/或其关联公司书面同意，任何人不得擅自使用、修改、复制、公开传播、改变、散布、发行或公开发表阿里云网站、产品程序或内容。此外，未经阿里云事先书面同意，任何人不得为了任何营销、广告、促销或其他目的使用、公布或复制阿里云的名称（包括但不限于单独为或以组合形式包含”阿里云”、Aliyun”、“万网”等阿里云和/或其关联公司品牌，上述品牌的附属标志及图案或任何类似公司名称、商号、商标、产品或服务名称、域名、图案标示、标志、标识或通过特定描述使第三方能够识别阿里云和/或其关联公司）。
6. 如若发现本文档存在任何错误，请与阿里云取得直接联系。

通用约定

格式	说明	样例
	该类警示信息将导致系统重大变更甚至故障，或者导致人身伤害等结果。	 **禁止：** 重置操作将丢失用户配置数据。
	该类警示信息可能导致系统重大变更甚至故障，或者导致人身伤害等结果。	 **警告：** 重启操作将导致业务中断，恢复业务所需时间约10分钟。
	用于补充说明、最佳实践、窍门等，不是用户必须了解的内容。	 **说明：** 您也可以通过按**Ctrl** + **A**选中全部文件。
>	多级菜单递进。	设置 > 网络 > 设置网络类型
粗体	表示按键、菜单、页面名称等UI元素。	单击 **确定**。
`courier` 字体	命令。	执行 `cd /d C:/windows` 命令，进入Windows系统文件夹。
斜体	表示参数、变量。	`bae log list --instanceid` *`Instance_ID`*
[]或者[a|b]	表示可选项，至多选择一个。	`ipconfig` *`[-all|-t]`*
{}或者{a|b}	表示必选项，至多选择一个。	`swich` *`{stand | slave}`*

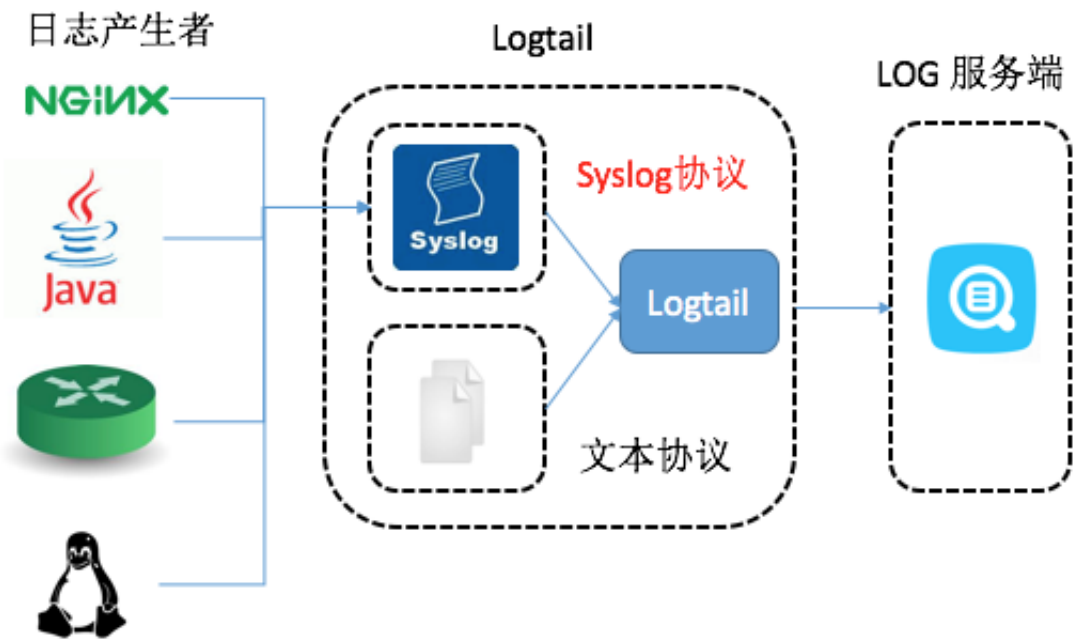
目录

1 隐藏文件夹

1.1 Syslog-采集参考

Logtail目前支持的接入端为syslog和文本文件，如下图所示：

图 **1-1: Logtail**支持的接入端



Logtail通过TCP协议支持syslog。配置Logtail采集syslog日志详细步骤请参见*Syslog*通过Logtail采集syslog日志。

syslog优势

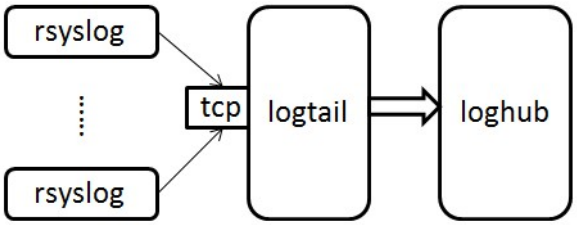
syslog概念请参考 *syslog*。

和利用文本文件相比，使用syslog时日志数据直接收集到LogHub，syslog不落盘、保密性好。免去了文件落盘和解析的代价，单机可达 80MB/S 吞吐率。

基本原理

Logtail 支持在本地配置TCP端口，接收syslog Agent转发的日志。Logtail开启TCP端口，接收rsyslog或者其他syslog Agent通过TCP协议转发过来的syslog数据，Logtail解析接收到的数据并转发到LogHub中。配置Logtail采集syslog日志过程请参见*Syslog*。Logtail、syslog、LogHub三者之间的关系如下图所示。

图 **1-2:** 基本原理



syslog日志格式

Logtail 通过 TCP 端口接收到的数据是流式的，如果要从流式的数据中解析出一条条的日志，日志格式必须满足以下条件：

- 每条日志之间使用换行符（`\n`）分隔，一条日志内部不可以出现换行符。
- 日志内部除了消息正文可以包含空格，其他字段不可以包含空格。

syslog日志格式如下：

```
$version $tag $unixtimestamp $ip [$user-defined-field-1 $user-defined-field-2 $user-defined-field-n] $msg\n"
```

各个字段含义为：

日志字段	含义
version	该日志格式的版本号，Logtail使用该版本号解析user-defined-field 字段。
tag	数据标签，用于寻找Project或Logstore，不可以包含空格和换行符。
unixtimestamp：	该条日志的时间戳。
ip	该条日志的对应的机器IP，如果日志中的该字段是 127.0.0.1，最终发往服务端的日志数据中该字段会被替换成TCP socket的对端地址。
user-defined-field	用户自定义字段，中括号表示是可选字段，可以有 0 个或多个，不可以包含空格和换行符。
msg	日志消息正文，不可以包含换行符，末尾的 `\n` 表示换行符。

以下示例日志即为满足格式要求的日志：

```
2.1 streamlog_tag 1455776661 10.101.166.127 ERROR com.alibaba.
streamlog.App.main(App.java:17) connection refused, retry
```

另外，不仅 syslog 日志可以接入Logtail，任何日志工具只要能满足以下条件，都可以接入：

- 可以将日志格式化，格式化之后的日志格式满足格式要求。
- 可以通过 TCP 协议将日志 append 到远端。

Logtail解析syslog日志规则

Logtail 需要增加配置以解析syslog日志。例如：

```
"streamlog_formats":
[
    {"version": "2.1", "fields": ["level", "method"]},
    {"version": "2.2", "fields": []},
    {"version": "2.3", "fields": ["pri-text", "app-name", "syslogtag
"]}
]
```

Logtail通过读取到的`version`字段到`streamlog_formats`中找到对应的`user-defined`字段的格式，应用该配置，上面的日志样例`version`字段为 2.1，包含两个自定义字段`level`和`method`，因此日志样例将被解析为如下格式：

```
{
    "source": "10.101.166.127",
    "time": 1455776661,
    "level": "ERROR",
    "method": "com.alibaba.streamlog.App.main(App.java:17)",
    "msg": "connection refused, retry"
}
```

`version`用于解析`user-defined`字段，tag用于寻找数据将要被发送到的Project或Logstore，这两个字段不会作为日志内容发送到阿里云日志服务。另外，Logtail预定义了一些日志格式，这些内置的格式都使用 0.1、1.1 这样以“0.”、“1.”开头的`version`值，所以用户自定义`version`不可以以“0.”、“1.”开头。

常见日志工具接入 Logtail syslog log

- **log4j**

  — 引入 log4j 库。

  ```
  <dependency>
        <groupId>org.apache.logging.log4j</groupId>
        <artifactId>log4j-api</artifactId>
        <version>2.5</version>
    </dependency>
  ```

```
  <dependency>
      <groupId>org.apache.logging.log4j</groupId>
      <artifactId>log4j-core</artifactId>
      <version>2.5</version>
  </dependency>
```

- 程序中引入 log4j 配置文件 `log4j_aliyun.xml`。

```
 <?xml version="1.0" encoding="UTF-8"?>
  <configuration status="OFF">
      <appenders>
          <Socket name="StreamLog" protocol="TCP" host="10.101.166
.173" port="11111">
              <PatternLayout pattern="%X{version} %X{tag} %d{UNIX}
 %X{ip} %-5p %l %enc{%m}%n" />
          </Socket>
      </appenders>
      <loggers>
          <root level="trace">
              <appender-ref ref="StreamLog" />
          </root>
      </loggers>
  </configuration>
```

其中 10.101.166.173:11111 是 Logtail 所在机器的地址。

- 程序中设置 ThreadContext。

```
package com.alibaba.streamlog;
  import org.apache.logging.log4j.LogManager;
  import org.apache.logging.log4j.Logger;
  import org.apache.logging.log4j.ThreadContext;
  public class App
  {
      private static Logger logger = LogManager.getLogger(App.
class);
      public static void main( String[] args ) throws Interrupte
dException
      {
          ThreadContext.put("version", "2.1");
          ThreadContext.put("tag", "streamlog_tag");
          ThreadContext.put("ip", "127.0.0.1");
          while(true)
          {
              logger.error("hello world");
              Thread.sleep(1000);
          }
          //ThreadContext.clearAll();
      }
  }
```

- **tengine**

  tengine 可以通过 syslog 接入 ilogtail。

  tengine 使用 `ngx_http_log_module`模块将日志打入本地 syslog agent，在本地 syslog agent 中 forward 到 rsyslog。

tengine 配置 syslog 请参考：*tengine配置syslog*

示例：

以 user 类型和 info 级别将 access log 发送给本机 Unix dgram(/dev/log)，并设置应用标记为 nginx。

```
access_log syslog:user:info:/var/log/nginx.sock:nginx
```

rsyslog 配置：

```
module(load="imuxsock") # needs to be done just once
input(type="imuxsock" Socket="/var/log/nginx.sock" CreatePath="on")
$template ALI_LOG_FMT,"2.3 streamlog_tag %timegenerated:::date-
unixtimestamp% %fromhost-ip% %pri-text% %app-name% %syslogtag% %msg
:::drop-last-lf%\n"
if $syslogtag == 'nginx' then @@10.101.166.173:11111;ALI_LOG_FMT
```

- **nginx**

  以收集 nginx accesslog 为例。

  access log 配置：

  ```
  access_log syslog:server=unix:/var/log/nginx.sock,nohostname,tag=
  nginx;
  ```

  rsyslog 配置：

  ```
  module(load="imuxsock") # needs to be done just once
  input(type="imuxsock" Socket="/var/log/nginx.sock" CreatePath="on")
  $template ALI_LOG_FMT,"2.3 streamlog_tag %timegenerated:::date-
  unixtimestamp% %fromhost-ip% %pri-text% %app-name% %syslogtag% %msg
  :::drop-last-lf%\n"
  if $syslogtag == 'nginx' then @@10.101.166.173:11111;ALI_LOG_FMT
  ```

  参考：*http://nginx.org/en/docs/syslog.html*

- **Python Syslog**

  示例：

  ```
  import logging
  import logging.handlers
  logger = logging.getLogger('myLogger')
  logger.setLevel(logging.INFO)
  #add handler to the logger using unix domain socket '/dev/log'
  handler = logging.handlers.SysLogHandler('/dev/log')
  #add formatter to the handler
  formatter = logging.Formatter('Python: { "loggerName":"%(name)s",
   "asciTime":"%(asctime)s", "pathName":"%(pathname)s", "logRecordC
  reationTime":"%(created)f", "functionName":"%(funcName)s", "levelNo
  ":"%(levelno)s", "lineNo":"%(lineno)d", "time":"%(msecs)d", "
  levelName":"%(levelname)s", "message":"%(message)s"}')
  ```

```
handler.formatter = formatter
logger.addHandler(handler)
logger.info("Test Message")
```

1.2 Syslog

Logtail支持在本地配置TCP端口，接收syslog Agent通过TCP协议转发过来的syslog数据，Logtail解析接收到的数据并转发至LogHub中。

前提条件

设置使用Logtail收集日志前，您需要安装Logtail。Logtail支持Windows和Linux两大操作系统，安装方法参见 *Linux* 和*Windows*。

步骤 1 在日志服务管理控制台创建Logtail syslog配置

1. 在日志服务云控制台单击目标项目，进入**Logstore**列表。
2. 选择目标Logstore，并单击**数据接入向导**图标，进入数据接入流程。
3. 选择数据源类型。

   单击**自定义数据**中的**Syslog**，并单击下一步。
4. 指定 Logtail配置的名称。

   配置名称只能包含小写字母、数字、连字符（-）和下划线（_），且必须以小写字母和数字开头和结尾，长度为 3~63 字节。

   > **说明：**
   > 配置名称设置后不可修改。
5. 填写**Tag**设置。

   如何设置Tag，请参考*Syslog-采集参考*。

图 1-3: 设置Tag

6. 酌情配置高级选项。

   请选择是否打开本地缓存。当日志服务不可用时，日志可以缓存在机器本地目录，服务恢复后进行续传。默认开启缓存，最大缓存值1GB。

7. 根据页面提示，应用Logtail配置到机器组。

   确认勾选所需的机器组并单击 应用到机器组 将配置应用到机器组。

   如果您还未创建机器组，需要先创建一个机器组。有关如何创建机器组，参见 创建*IP*地址机器组。

图 1-4: 应用到机器组

步骤 2 配置Logtail使协议生效

从机器Logtail安装目录下找到 `ilogtail_config.json`，一般在 `/usr/local/ilogtail/`目录下面。根据需求修改和syslog相关的配置。

1. 确认syslog功能已开启。

true表示syslog功能处于打开状态，false表示关闭状态。

```
"streamlog_open" : true
```

2. 配置syslog用于接收日志的内存池大小。程序启动时会一次性申请指定大小的内存，请根据机器内存大小以及实际需求填写，单位是MB。

```
"streamlog_pool_size_in_mb" : 50
```

3. 配置缓冲区大小。需要配置Logtail每次调用socket io rcv 接口使用的缓冲区大小，单位是byte。

```
"streamlog_rcv_size_each_call" : 1024
```

4. 配置日志syslog格式。

```
"streamlog_formats":[]
```

5. 配置TCP绑定地址和端口。需要配置Logtail用于接收syslog日志的TCP绑定地址和端口，默认是绑定0.0.0.0下的11111端口。

```
"streamlog_tcp_addr" : "0.0.0.0",
"streamlog_tcp_port" : 11111
```

6. 配置完成后重启Logtail。重启Logtail要执行以下命令关闭Logtail客户端，并再次打开。

```
sudo /etc/init.d/ilogtaild stop
sudo /etc/init.d/ilogtaild start
```

步骤 3 安装rsyslog并修改配置

如果机器已经安装rsyslog，忽略这一步。

1. 安装rsyslog。

   安装方法请参见：

   - *Ubuntu 安装方法*
   - *Debian 安装方法*
   - *RHEL/CENTOS 安装方法*

2. 修改配置。

   在 `/etc/rsyslog.conf` 中根据需要修改配置，例如：

```
$WorkDirectory /var/spool/rsyslog # where to place spool files
$ActionQueueFileName fwdRule1 # unique name prefix for spool files
$ActionQueueMaxDiskSpace 1g # 1gb space limit (use as much as
possible)
$ActionQueueSaveOnShutdown on # save messages to disk on shutdown
$ActionQueueType LinkedList # run asynchronously
```

```
$ActionResumeRetryCount -1 # infinite retries if host is down
# 定义日志数据的字段
$template ALI_LOG_FMT,"0.1 sys_tag %timegenerated:::date-unixtimest
amp% %fromhost-ip% %hostname% %pri-text% %protocol-version% %app-
name% %procid% %msgid% %msg:::drop-last-lf%\n"
*.* @@10.101.166.173:11111;ALI_LOG_FMT
```

说明：

模板 `ALI_LOG_FMT` 中第二个域的值是 `sys_tag`，这个取值必须和步骤 1 中创建的一致，这个配置的含义是将本机接收到的所有（`\*.\*`）syslog 日志按照 `ALI_LOG_FMT`格式化，使用 TCP 协议转发到 10.101.166.173:11111。机器 10.101.166.173 必须在步骤 1 中的机器组中并且按照步骤 2 配置。

3. 启动rsyslog。

```
sudo /etc/init.d/rsyslog restart
```

启动之前请先检查机器上是否安装了其他syslog的Agent，比如 syslogd、sysklogd、syslog-ng 等，如果有的话请关闭。

上面三步完成之后就可以将机器上的syslog收集到日志服务了。

更多信息

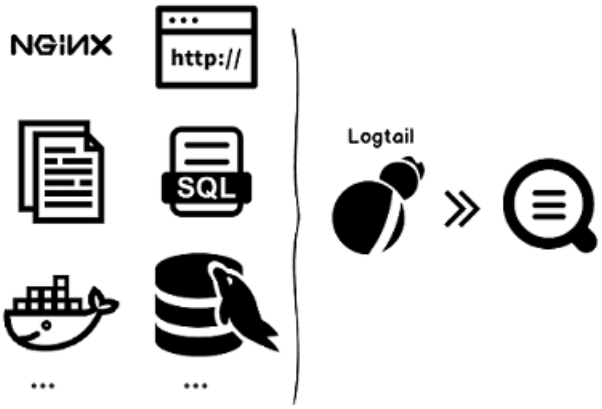
有关syslog日志采集的更多信息以及如何格式化syslog数据，请参见*Syslog-采集参考*。

2 Logtail采集

2.1 简介

Logtail接入服务是日志服务提供的日志采集Agent，通过控制台方式帮助您实时采集阿里云ECS、自建IDC、其他云厂商等服务器上的日志。

图 **2-1: Logtail采集功能**



功能优势

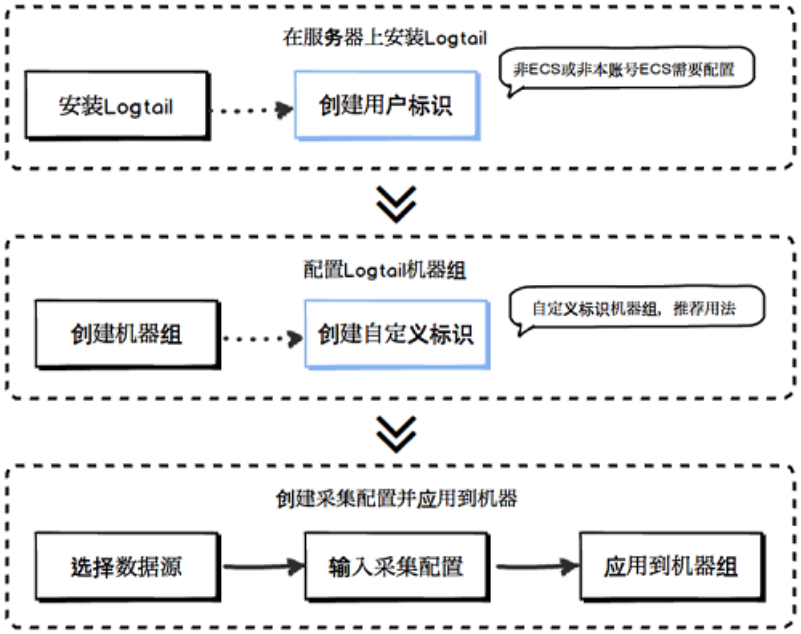
- 基于日志文件、无侵入式的收集日志。用户无需修改应用程序代码，且日志收集不会影响用户应用程序的运行逻辑。
- 除支持文本日志采集外，还支持binlog、http、容器stdout等采集方式。
- 对于容器支持友好，支持标准容器、swarm集群、Kubernetes集群等容器集群的数据采集。
- 能够稳定地处理日志收集过程中各种异常。当遇到网络异常、服务端异常等问题时会采用主动重试、本地缓存数据等措施保障数据安全。
- 基于服务端的集中管理能力。用户在安装Logtail后（参见 *Windows* 和 *Linux* ），只需要在服务端集中配置需要收集的机器、收集方式等信息即可，无需逐个登录服务器进行配置。
- 完善的自我保护机制。为保证运行在客户机器上的收集Agent不会明显影响用户自身服务的性能，Logtail客户端在CPU、内存及网络使用方面都做了严格的限制和保护机制。

处理能力与限制

参见相关限制说明。

配置流程

图 2-2: 配置流程



通过Logtail采集服务器日志可以通过以下步骤完成：

1. 安装Logtail。在需要采集日志的源服务器上安装Logtail操作请参见*Windows* 和 *Linux*。
2. 创建用户自定义标识机器组。从阿里云ECS采集日志不需要执行此步骤。
3. 创建*IP*地址机器组。日志服务通过机器组的方式管理所有需要通过Logtail客户端采集日志的服务器。日志服务支持通过IP或者自定义标识的方式定义机器组。您也可以在应用Logtail配置到机器组时，根据提示创建机器组。
4. 创建Logtail采集配置，并应用到机器组。您可以通过**数据接入向导**创建Logtail配置以 文本日志、*Syslog*等，并将该Logtail配置应用到机器组。

在完成如上流程后，您的ECS服务器上需要收集的新增日志会被主动收集、发送到对应Logstore中，历史数据不会被收集。您可以通过日志服务控制台或者SDK及API查询到这些日志。您还可以通过日志服务查询到所有ECS服务器上的Logtail收集日志状态，例如是否在正常收集，是否有错误等。

Logtail接入服务在日志服务控制台上的完整操作请参考*Logtail* 收集日志 。

容器

- 阿里云容器服务Swarm：参见集成日志服务。

- 阿里云容器服务Kubernetes：参见采集*Kubernetes*日志
- 自建Kubernetes：参见自建*Kubernetes*安装方式
- 自建其他Docker集群：参见标准*Docker*日志采集

核心概念

- **机器组**：一个机器组包含一或多台需要收集一类日志的机器。通过绑定Logtail配置到机器组，可以让日志服务根据同样的Logtail配置采集一个机器组内所有服务器上的日志。您也可以通过日志服务控制台方便地对机器组进行管理（包括创建、删除机器组，添加、移除机器等）。同一个机器组内不可同时包含Windows和Linux机器，但可以包含不同版本的Windows Server或者不同发行版本的Linux机器。
- **Logtail客户端**：Logtail是运行在需要收集日志的服务器上上执行日志收集工作的Agent。安装步骤请参考 *Windows* 和 *Linux*。 在服务器上安装Logtail后，需要配置Logtail并应用到机器组。
  - **Linux** 下，Logtail安装在 `/usr/local/ilogtail` 目录下，并启动两个以 ilogtail 开头的个独立进程，一个为收集进程，另外一个为守护进程，程序运行日志为 `/usr/local/ilogtail/ilogtail.LOG`。
  - **Windows** 下，Logtail安装在目录 `C:\Program Files\Alibaba\Logtail`（32 位系统）或 `C:\Program Files (x86)\Alibaba\Logtail`（64 位系统）下。您可以通过Windows管理工具>服务查看到两个Windows Service，LogtailWorker负责收集日志，LogtailDaemon负责守护工作程序。程序运行日志为安装目录下的 `logtail_*.log`。
- **Logtail配置**：是Logtail收集日志的策略集合。通过为Logtail配置数据源、收集模式等参数，来对机器组内所有服务器进行定制化的收集策略。Logtail配置定义了如何在机器上收集一类日志并解析、发送到日志服务的指定日志库。您可以通过控制台对每个Logstore添加Logtail配置，表示该Logstore接收以此Logtail配置收集的日志。

基本功能

Logtail接入服务提供如下功能：

- **实时收集日志**：动态监控日志文件，实时地读取、解析增量日志。日志从生成到发往服务端的延迟一般在3秒内。

> **说明：**
> Logtail接入服务不支持对历史数据的收集。对于一条日志，读取该日志的时刻减去日志产生的时刻，差值超过5分钟的会被丢弃。

- **自动处理日志轮转**：很多应用会按照文件大小或者日期对日志文件进行轮转（rotation），把原日志文件重命名，并新建一个空日志文件等待写入。例如：监控*app.LOG*，日志轮转会产生*app.LOG.1*，*app.LOG.2* 等。您可以指定收集日志写入的文件，如 *app.LOG*，Logtail会自动检测到日志轮转过程，保证这个过程中不会出现日志数据丢失。
- **多种采集输入源**：Logtail除支持文本日志采集外，还支持syslog、http、MySQL binlog等输入源，更多内容参见采集数据源配置章节。
- **兼容开源采集Agent**：Logtail支持Logstash、Beats等开源软件采集的数据作为输入源，更多内容参见采集数据源配置章节。
- **自动处理收集异常**：因为服务端错误、网络措施、Quota超限等各种异常导致数据发送失败，Logtail会按场景主动重试。如果重试失败则会将数据写入本地缓存，稍后自动重发。
- **灵活配置收集策略**：可以通过Logtail配置来非常灵活地指定如何在一台ECS服务器上收集日志。具体来说，您可以根据实际场景选择日志目录、文件，既可精确匹配，也可通过通配符模糊匹配。您可以自定义日志收集提取的方式和各个提取字段的名称，日志服务支持正则表达式方式的日志提取。另外，由于日志服务日志数据模型要求每条日志必须有精确的时间戳信息，Logtail提供了自定义的日志时间格式，方便您从不同格式的日志数据中提取必须要的日志时间戳信息。
- **自动同步收集配置**：您在日志服务控制台上新建或更新配置，Logtail一般在3分钟时间内即可自动接受并使之生效，更新配置过程中数据收集不丢失。
- **自动升级客户端**：在您手动安装Logtail到服务器后，日志服务负责Logtail 自动运维升级，此过程无需您参与。在整个Logtail升级过程中日志数据不丢失。
- **自我监控状态**：为避免Logtail客户端消耗您太多资源而影响您其他服务。Logtail客户端会实时监控自身CPU和内存消耗。如果Logtail客户端在运行过程中，资源使用超出限制将会自动重启，避免影响机器上的其它作业。同时，该客户端也会有主动的网络限流保护措施，防止过度消耗用户带宽。
- **签名数据发送**：为保证您的数据在发送过程中不会被篡改，Logtail客户端会主动获取用户的阿里云访问秘钥并对所有发送日志的数据包进行数据签名。

说明：

Logtail客户端在获取您的阿里云访问秘钥时采用HTTPS通道，保障您的访问秘钥安全性。

2.2 选择网络

采集日志数据到日志服务时，日志数据可以通过阿里云内网、公网和全球加速网络传输。

网络类型

- 公网：使用公网传输日志数据，不仅会受到网络带宽的限制，还可能会因网络抖动、延迟、丢包等影响数据采集的速度和稳定性。
- 阿里云内网：阿里云内网为千兆共享网络，日志数据通过阿里云内网传输比公网传输更快速、稳定。内网包括**VPC**环境和经典网络环境。
- 全球加速：利用阿里云CDN边缘节点进行日志采集加速，相对公网采集在网络延迟、稳定性上具有很大优势。

如何选择网络

- 内网：

  您的日志数据是否通过阿里云内网传输，取决于您的服务器类型以及服务器和日志服务Project是否在同一地域。仅有以下两种情况可以使用阿里云内网传输：

  - 本账号下的**ECS**和日志服务**Project**在同一地域。
  - 其他账号的**ECS**和本账号的日志服务**Project**在同一地域。

  因此，建议您在ECS的相同地域下创建日志服务Project，并将日志采集到同地域的日志服务Project中。ECS上的日志数据自动通过阿里云内网写入日志服务，不消耗公网带宽。

  > 说明：
  > 在服务器上安装Logtail时，选择的地域必须和Project所在地域一致，否则无法正常采集日志数据。

- 全球加速：

  如果您的服务器分布在海外各地的自建机房、或者来自海外云厂商，使用公网传输数据可能会出现网络延迟高、传输不稳定等问题，可以通过全球加速传输数据。全球加速利用阿里云CDN边缘节点进行日志采集加速，相对公网采集在网络延迟、稳定性上具有很大优势。

- 公网：

  在以下两种情况时，您可以选择网络类型为公网：

  - 服务器为ECS，但和日志服务Project位于不同地域。
  - 服务器为其他云厂商服务器、自建IDC。

服务器类型	是否与Project同一地域	是否需要配置AliUid	网络类型
本账号下的ECS	同一地域	不需要	阿里云内网
	不同地域	不需要	公网或全球加速
其他账号下的ECS	同一地域	需要	阿里云内网
	不同地域	需要	公网或全球加速
其他云厂商服务器、自建IDC	-	需要	公网或全球加速

说明：

日志服务无法获取非本账号下ECS、其他服务器的属主信息，请在安装Logtail后手动配置用户标识（AliUid），否则安装Logtail的服务器会心跳异常、无法收集日志。详细步骤请参见 为非本账号*ECS*、自建*IDC*配置*AliUid*。

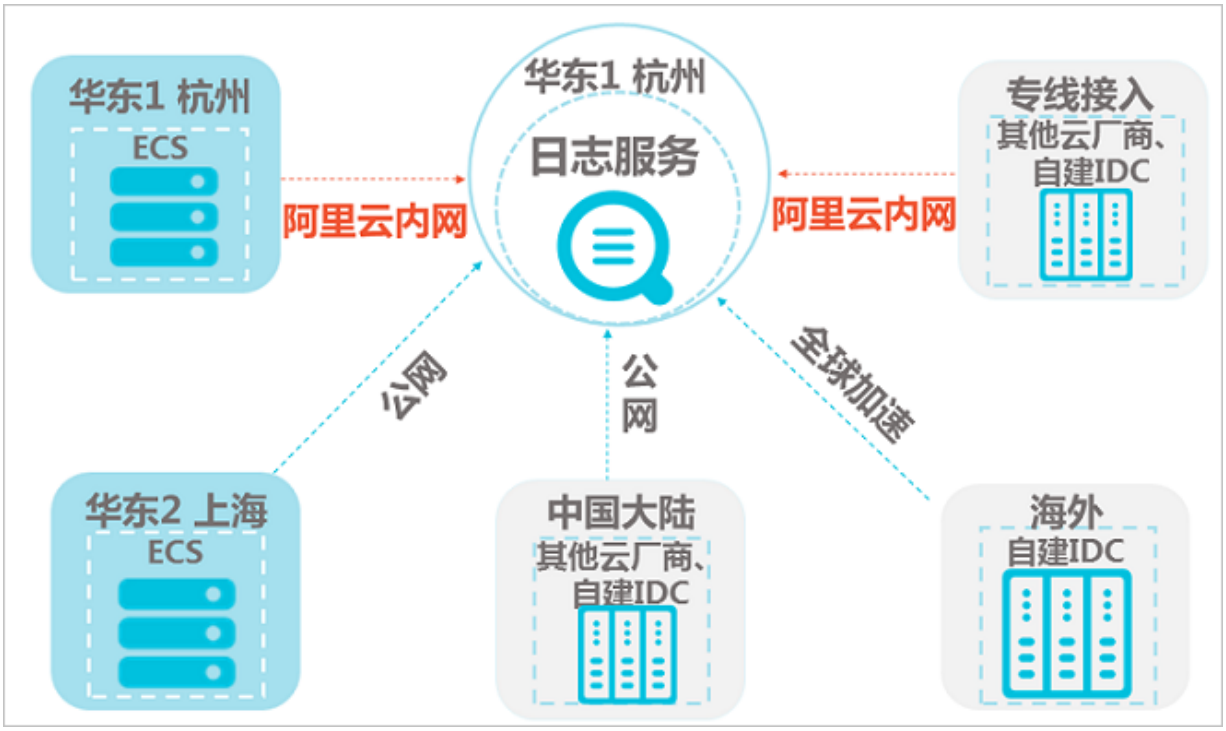
网络选择示例

以下是各种常见场景的网络选择示例。

场景类型	日志服务Project地域	服务器类型	ECS地域	安装Logtail时选择的地域	网络类型	是否需要配置AliUid
相同地域场景	华东1（杭州）	本账号ECS	华东1（杭州）	华东1（杭州）	内网	不需要
不同地域场景	华东2（上海）	本账号ECS	华北1（北京）	华北1（北京）	公网	不需要
其他账号场景	华东2（上海）	其他账号ECS	华北1（北京）	华北1（北京）	公网	需要
本地机房场景	华东5（深圳）	自建IDC	-	华东5（深圳）	公网	需要
全球加速场景	香港	自建IDC	-	香港	全球加速	需要

说明：

全球加速场景中，日志服务Project创建在香港地域，服务器为全球各地的自建机房，数据采集的速度和可靠性尤为重要，所以建议您在类似场景下安装Logtail时选择香港地域的全球加速网络类型。日志数据通过全球加速传输，比公网传输的网络稳定性更高、性能更好。



经典网络切换VPC后的配置更新

安装Logtail后，如果您的ECS网络类型由经典网络切换为VPC，请参考以下步骤更新配置。

1. 以管理员身份重启Logtail。
   - **Linux**：

     ```
     sudo /etc/init.d/ilogtaild stop
     sudo /etc/init.d/ilogtaild start
     ```

   - **Windows**：

     打开控制面板中的管理工具，打开服务，找到*LogtailWorker*并右键单击重新启动。
2. 更新机器组配置。
   - 自定义标识

     若机器组中配置了自定义标识，则无需手动更新机器组配置，可以直接正常使用VPC网络。
   - **IP地址**

若机器组中配置了ECS云服务器IP地址，则需将机器组内的IP更换为重启Logtail后获取的IP地址，即*app_info.json*中的ip字段。

*app_info.json*文件地址：

- Linux：*/usr/local/ilogtail/app_info.json*
- Windows x64：*C:\Program Files (x86)\Alibaba\Logtail\app_info.json*
- Windows x32：*C:\Program Files\Alibaba\Logtail\app_info.json*

2.3 相关限制说明

表 2-1: 文件采集限制

分类	限制说明
文件编码	支持UTF8/GBK编码日志文件，建议使用UTF8编码以获得更好的处理性能。如果日志文件为其它编码格式则会出现乱码、数据丢失等错误。
日志文件大小	无限制。
日志文件轮转	支持，流转文件名支持配置为`.log*`或者`.log`。
日志解析阻塞时采集行为	日志解析阻塞时，Logtail会将该日志文件FD保持打开状态；若解析阻塞期间出现多次日志文件轮转，Logtail会尽可能保持各个轮转日志解析顺序。若未解析的日志轮转超过20个，则后续文件不被处理。更多内容请参考相关技术文章。
软链接	支持监控目录为软链接。
单条日志大小	单条日志大小限制为512KB。多行日志按行首正则表达式划分后，每条日志大小限制仍为512KB。若日志超过512KB后，会强制拆分多块进行采集。例如：日志单条1025KB，则第一次处理前512KB，第二次处理512KB，第三次处理1KB。
正则表达式	正则表达式类型支持Perl兼容正则表达式。
同一文件对应多个采集配置	不支持，建议文件采集到一个Logstore，可以配置多份订阅。若有相关需求，可通过为文件配置软连接的方式绕过该限制。

分类	限制说明
文件打开行为	Logtail会保持被采集文件处于打开状态，若该文件超过5分钟未修改，则会关闭该文件（未发生轮转情况下）。
首次日志采集行为	Logtail只采集增量的日志文件，首次发现文件修改后，若文件大小超过1M，则从最后1M处开始采集，否则从开始位置采集；若配置下发后日志文件一直无修改，则不采集该文件。
非标准文本日志	对于日志中包含'\0'的行，该条日志会被截断到第一个'\0'处。

表 2-2: Checkpoint管理

项目	能力与限制
Checkpoint超时时间	若文件超过30天未修改，则会删除该Checkpoint。
Checkpoint保存策略	定期保存（15分钟），程序退出时会自动保存。
Checkpoint保存位置	保存路径默认为`/tmp/logtail_checkpoint`，可根据 配置启动参数 调整参数。

表 2-3: 配置限制

项目	能力与限制
配置更新	用户的配置更新生效的延时约30秒。
配置动态加载	支持，且其中某一配置更新不影响其他采集。
配置数	理论无限制，建议一台服务器采集配置数不超过100。
多租户隔离	各个采集配置间隔离。

表 2-4: 资源、性能限制

项目	能力与限制
日志处理吞吐能力	原始日志流量默认限制为2MB/s（数据会编码压缩后上传，一般压缩率为5-10倍）。超过该日志流量则有可能丢失日志，可根据 配置启动参数 调整参数。

项目	能力与限制
最大性能	单核能力：极简模式日志最大处理能力为100MB/s，正则默认最大处理能力为20MB/s（和正则复杂度有关），分隔符日志最大处理能力为40MB/s，JSON日志最大处理能力为30MB/s；开启多个处理线程性能可提高1.5-3倍左右
监控目录数	主动限制监控的目录层深，避免出现过多消耗用户资源。如果监控上限已到，则放弃监控更多目录和日志文件。限制最多3000个目录（含子目录）。
默认资源限制	默认Logtail最多会占用40%CPU、256MB内存，如日志产生速率较高，可根据 配置启动参数 调整参数。
资源超限处理策略	若3分钟内Logtail占用的相关资源超过最大限制，则Logtail会强制重启，此时数据可能会丢失或重复。

表 2-5: 错误处理限制

项目	能力与限制
网络错误处理	在出现网络异常时会主动重试并自动调整重试间隔。
资源配额超限处理	若数据发送速率超出Logstore最大配额，Logtail会阻塞采集并自动重试。。
超时最大尝试时间	若数据持续发送失败超过6小时，则丢弃该数据。
状态自检	支持异常情况下自动重启，例如程序异常退出及使用资源超限等。

表 2-6: 其他限制

项目	能力与限制
日志采集延迟	正常情况下从日志flush磁盘到Logtail采集改日志延迟不超过1秒（阻塞状态下除外）。
日志上传策略	Logtail会将同一文件的日志自动聚合上传，聚合条件为：日志超过2000条、日志总大小超过2M或者日志采集时间超过3秒，任一条件满足则触发上传行为。