

Research and Realization of Monitoring and Alarming System of Ship Engine Room Based on “Cloud”

Shouhan Pei

Nanjing Saibao Institute of Industrial Technology, Nanjing, Jiangsu, 201908, China

Abstract

A design scheme of monitoring and alarming system for ship engine room based on cloud is proposed. By writing the data communication protocol, Tiantong No.1 satellite communication system is adopted for data transmission of shipborne Web server, and the “cloud” architecture server is studied, the “cloud” server architecture is designed, and the algorithm is used to realize the collaborative work of multiple servers. The final results show that the communication mode of the monitoring and alarm system based on “cloud” is reliable.

Keywords

communication; “Cloud”; Engine room monitoring; Platform

基于“云”的船舶机舱监测报警系统的研究与实现

裴守含

南京赛宝工业技术研究院, 中国·江苏 南京 201908

摘要

提出了一种基于“云”的船舶机舱监测报警系统的设计方案;通过编写数据通信协议,采用天通一号卫星通信系统进行船载 Web 服务器的数据传输,并对“云”架构服务器进行研究,设计“云”服务器架构,利用了算法实现多服务器协同工作;最终结果表明,基于“云”的船舶机舱监测报警系统的通信方式可靠。

关键词

通信;“云”;机舱监测;平台

1 引言

近年来,以互联网信息化、云计算技术、大数据分析以及人工智能为代表的新一轮技术正经历高速发展时期,船舶行业在这些技术的推动下逐渐走向智慧化的道路。同时这些技术带着船舶机舱监测报警系统的发展。

船舶机舱监测报警系统的发展历程按照系统技术架构的不同可分为三大阶段,分别为 C/S 架构的系统阶段, B/S 架构的系统阶段以及“云”架构的系统阶段。文献^[1]介绍了以 C/S 架构的英国 TANSAS 公司的 Fleet View 系统与挪威 C.MAP 公司的 Ocean View 系统;文献^[2]介绍了以 B/S 架构的 Kongsberg 公司的 K-Chief 600 系统;文献^[3]介绍了以“云”架构的海兰信公司的 VMS 系统。无论是 C/S 架构的机舱监测报警系统,还是以 B/S 架构的机舱监测报警系统都只局限于船舶上,无法将船舶机舱监测报警系统延伸至岸基,同时现

有的“云”架构系统不能实现船上已有机舱监测报警系统的船岸信息一体化。

本文通过提出基于“云”的船舶机舱监测报警系统,利用天通一号卫星传输船载终端的监控数据,实现多服务器协同工作,最终以云平台的方式展示监测的内容。基于“云”的思想去建设共享型和智慧型的机舱监测报警系统,可以充分利用与享受岸基资源,使岸基与船端信息交互能力得以提升,对只局限于船舶上的机舱监测报警系统来说,具有开拓性的意义。

2 系统设计方案

基于“云”的船舶机舱监测报警系统是一套从船载 Web 服务器进行数据抓取,利用天通一号卫星移动通信系统将抓取的数据传输到岸基,并存储在岸基“云”架构的服务器中,最后通过云平台将抓取数据表现出来的多种技术相结合的先

进系统。其总架构设计如图1所示,主要包含三个部分,第一部分为数据抓取,通过设计抓取数据包程序实现数据的实时传输;第二部分为数据存储,采用“云”架构技术,通过“云”架构的设计和部署,它不仅拥有多服务器协同工作的功能,而且能够确保所存储数据的安全性和隐私性。同时,“云”架构实现服务器负载均衡,大大提高了服务器的使用率;第三部分为数据展示,数据展示利用云平台。用户可以随时随地进行相关机舱数据查看和数据报警功能的查看,实现机舱状态远程实时监测,同时大大降低船舶的安全隐患和保障船员的生命安全。

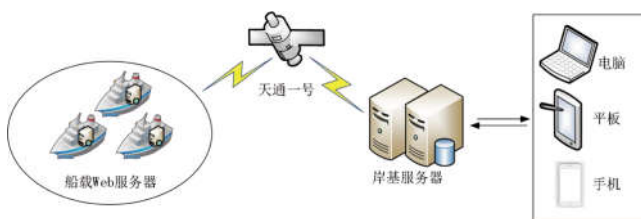


图1 基于“云”的船舶机舱监测报警系统结构图

3 系统相关研究

3.1 通信研究

为了实现数据准确地传输,以及不破坏船上原有的监测系统。通过重新在船舶上装入一台服务器并与船载 Web 服务器建立局域网从而实现船载 Web 服务器的数据共享。当数据共享设置成功后,利用加装的服务器来部署抓取程序。抓取程序中包含发送程序,能够利用船上已有的天通一号卫星移动通信系统的通信设备进行数据的传输。同时发送程序还能将抓取的数据准确传输到岸基“云”架构的服务器中。在岸基“云”架构的服务器中部署的接收程序和解析程序能够不断地侦测、接收和解析所接收到的数据,然后存储于“云”架构的服务器中,从而实现数据的完整传输。数据传输的通信设计如图2所示。

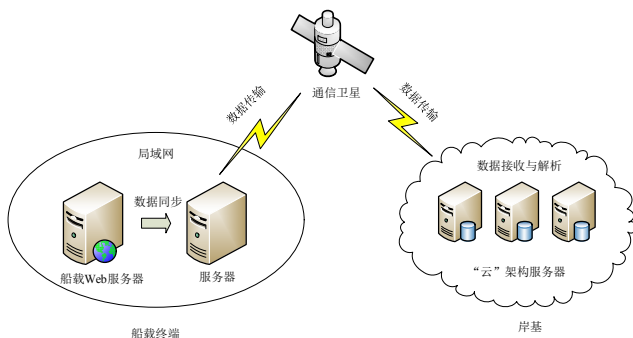


图2 数据通信设计

3.2 多服务器协同研究

由于船舶抓取的数据量越来越多,使用云平台的用户逐渐增多。单台服务器已不能满足日益增长的各种服务需求。因此,有必要通过采用多服务器协同工作的方法来解决这些问题。本系统通过使用反向代理负载均衡技术从而实现多服务器协同工作。反向代理负载均衡是指使用反向代理服务器(Nginx 服务器)来实现服务器负载均衡^[4]。当用户浏览器发出访问请求时,此请求会先被反向代理服务器接收,当反向代理服务器在接收请求后会根据反向代理服务器中编写的程序和负载均衡算法将请求转发给不同的 Web 服务器去处理。

反向代理服务器中的关键程序举例说明如下:

定义负载均衡的服务器的 IP 及服务器状态

```
Upstream myServer {  
    Server 192.168.1.5:8080 down;  
    Server 192.168.1.6:8081 weight=2;  
    Server 192.168.1.7:8082;  
    Server 192.168.1.8:8083 backup;}
```

在需要使用负载的 Server 节点下添加
proxy_pass http://myServer;

其中 down 表示当前此 IP 地址的 Web 服务器暂时不参与负载; weight 表示默认值为 1,当 weight 的值越大时表示此 IP 地址的服务器负载的权重越大,即此 Web 服务器访问的次数越多; backup 表示替补的 Web 服务器,当其它所有的非 backup 服务器死机或者忙的时候,则请求此 IP 地址的服务器。proxy_pass 表示用户可以直接访问的 Nginx 服务器的地址名称。Upstream 表示配置负载均衡。

反向代理服务器在向不同 Web 服务器转发请求的过程中用到负载均衡算法进行请求分配。该算法是反向代理服务器中自带的加权轮询算法,它是一项经典的负载均衡算法,位于 Nginx 的负载均衡模块 ngx_http_upstream_module 模块里的 ngx_http_upstream_round_robin.c 文件中^[5]。算法中包含四个关键的回调函数,分别为 us->peer.init 初始化函数、r->upstream->peer.get 选择初始化函数、ngx_http_upstream_get_round_robin_peer 计算并选择最大权值函数以及 ngx_http_upstream_free_round_robin_peer 释放函数。us->peer.init 初始化函数的作用是根据 upstream 里的各个 Server 配置项做初始准备工作,配置文件解析完不再调用;r->upstream->peer.

get 选择初始化函数的作用是选择合适的 Web 服务器做准备工作; ngx_http_upstream_get_round_robin_peer 计算并选择最大权值函数的作用是完成选择合适的 Web 服务器; ngx_http_upstream_free_round_robin_peer 释放函数在完成反向代理服务器和 Web 服务器之间的交互后都会被调用。四个回调函数在算法执行过程中的逻辑关系如下图 3 所示。

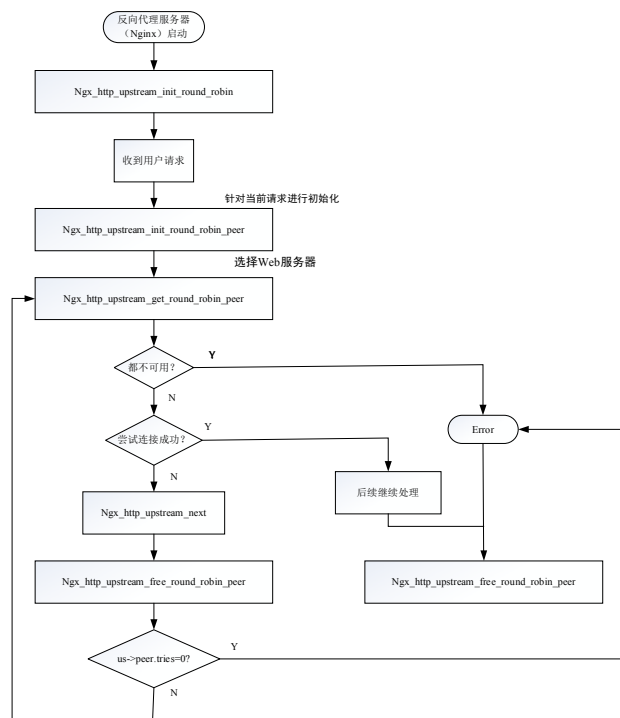


图 3 反向代理服务器加权轮询算法流程

3.3 “云”服务器架构研究

船舶机舱监测报警系统中的服务器主要功能大致可以分为接收数据、解析数据、存储数据、数据备份、数据调用以及提供用户可视化界面访问等功能。由于注册的船舶用户与日俱增，相应地机舱监测数据也会爆发式的增长以及考虑到数据的隐私性与安全性，机舱监测报警系统的可视化平台中的服务器则采用“云”服务器架构的方式，将抓取到的数据存放“云”中。

“云”服务器架构是通过部署的私有云架构与阿里云结合而实现。主要包括反向代理服务器、数据库服务器、业务服务器及 Web 服务器。反向代理服务器用于转发用户的请求; Web 服务器有多个，用于响应用户的请求并提供“云”架构机舱监测报警系统可视化平台的浏览服务; 数据库服务器用于对收集的数据进行存储; 业务服务器用于实现可视化平台

的业务逻辑。具体的“云”服务器架构如图 4 所示。

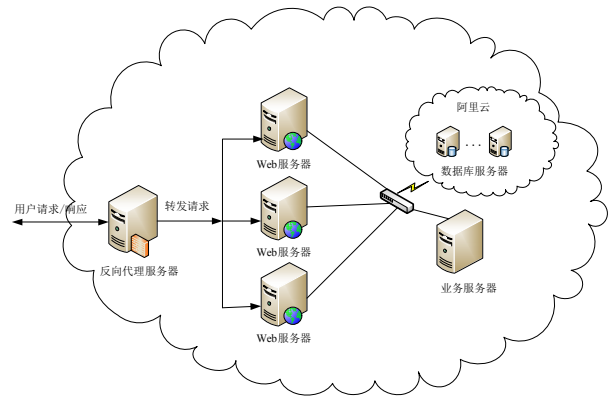


图 4 “云”服务器架构

4 系统平台实现

通过采用云平台，用户使用带浏览器的手机、电脑、平板等设备即可随时随地访问本系统。系统的主要功能为船舶机舱监测与报警功能，其中机舱监测的相关数据由界面直接显示; 报警功能则是通过监测机舱的主要设备，利用设备运行的高低限值来提供报警功能。当报警存在时，会具体到某艘船舶的机舱所监测的名称，并在系统的监测页面中具体的一项出现告警状态，且此项告警会以不同的颜色显示。实现此功能的关键代码如下:

```

    @RequestMapping("/api/shipAlarms")
    public class ShipAlarmController extends ShipBaseGrid-
    Controller<ShipAlarm, Long> {
        @Autowired
        private ShipAlarmService shipAlarmService;
        @Override
        protected EntityCRUDService<ShipAlarm, Long> getEn-
        tityService() {
            return shipAlarmService;
        }
        @RequestMapping(method = RequestMethod.GET,
        params = {"pageNo"})
        public Page<ShipAlarm> query() {
            Page result = super.query();
            return result;
        }
    }
  
```

```
@RequestMapping(value =("/{id}", method = Request-  
Method.GET)  
public ShipAlarm get(@PathVariable("id") Long id) {  
    ShipAlarm ship = super.get(id);  
    return ship;  
}  
  
@RequestMapping(value =("/{id}/handle", method =  
RequestMethod.PUT)  
public boolean handle(@PathVariable("id") Long id,  
@RequestBody AlarmHandleResultVo resultVo) {  
    Integer handleResult = resultVo.getHandleResult();  
    AlarmHandleResult alarmHandleResult =  
AlarmHandleResult.valueOfOrdinal(handleResult);    shipAl-  
armService.handle(id,alarmHandleResult,resultVo.getHandleDe-  
scrip());  
    return true;  
}
```

船舶机舱监测报警系统的界面如图5所示。

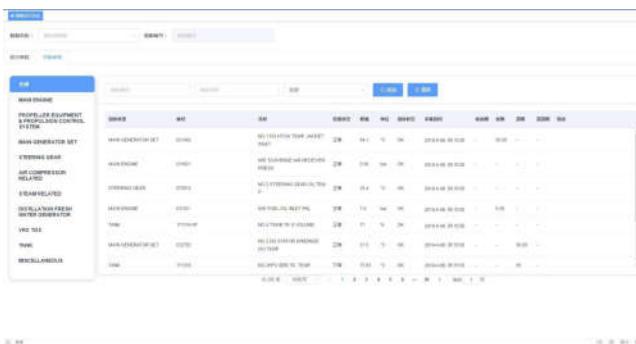


图5 船舶机舱监测报警界面

5 结语

基于“云”的船舶机舱监测报警系统是集计算机技术、通信技术、“云”服务器部署技术以及互联网技术等技术于一体、运用 SQL 语句、Java 语言和网页脚本语言等语言，是涉及到多个先进技术领域的监测报警系统。它的实现旨在为从事船舶行业的人服务的宗旨，将船员的生命放在第一位，避免人员伤亡和财产损失，同时为船员和船东在管理方面省时省力，对实现船岸一体化与无人机舱具有重要的研究价值。

参考文献

- [1] 刘洪卫. 基于 B/S 模式的岸基船舶监控系统的设计与实现 [D]. 哈尔滨: 哈尔滨工程大学, 2012.
- [2] 学牛. K-Chief 600 船舶自动化系统 [J]. 热能动力工程, 2011, 26(06): 704.
- [3] 桂雪琴. 遵循商业本质打造国际竞争优势——访北京海兰信数据科技股份有限公司董事长申万秋 [J]. 船舶物资与市场, 2016(02): 16-19.
- [4] 孔祥真, 张丁, 李忠远. Linux 负载均衡集群技术在网络服务器中的应用 [J]. 软件导刊, 2016, 15(12): 144-147.
- [5] 王利萍. 基于 Nginx 服务器集群负载均衡技术的研究与改进 [D]. 济南: 山东大学, 2015.