Research on Realization of Communication Method of Data Acquisition and Distribution Platform Based on Multiple Transmission Protocols

Zhao Long [1], Yushuang Ma [1] , Rongjin Yang [3] , Xiuhong Li [1,2+] , Song Qiao[1] , Zhenwei Song[1] and

Zhang Yi [1]

[1] State Key Laboratory of Remote Sensing Science, College of Global Change and Earth System Science, Beijing Normal University, Beijing 100875, China

[2] Joint Center for Global Change Studies, Beijing 100875, China

[3] Chinese Research Academy of Environmental Sciences, No.8, Da Yang Fang, An Wai, Chao Yang, Beijing, 100012, China.

Abstract. This paper mainly introduces the advantages and disadvantages of adopting different data transmission protocols in order to realize data communication in the data acquisition and distribution platform of wireless sensor network. The TCP connections based on IPv4 protocol and IPv6 protocol and the transmission efficiency of the HTTP-based stateless transfer protocol in practical applications are studied and compared in the corresponding application scenarios. The transmission header based on the IPv4 protocol is more complicated than IPv6. The IPv6 address space is huge and the terminal address allocation of the mass terminal can be realized. TCP connections maintain two-way communication between the client and the server, but consume too much for the server, while stateless protocol HTTP consumes less for the server and its disadvantage is the inability to record two-way communications.

Keywords: TCP, IPv4, IPv6, HTTP, WSN

1. Introduction

Modern Internet of things technology is developing rapidly. Terminal and server-side communication technology is constantly updated. Commonly used in the client and server-side communication protocols include IPv4, IPv6[1], [2] and HTTP (Hypertext Transfer Protocol)[3]. Gao Song designed and implemented an embedded remote monitoring system based on HTTP protocol. The system adopts B / S (Browser / Server) architecture to realize real-time monitoring of user data[4], Liu Zunmin uses the TCP protocol acquisition of simulated floating vehicles and traffic flow of objective information data at the same time through the HTTP protocol to upload real-time traffic congestion subjective information data[5], Chen Rongchao designed to achieve a TCP Socket and HTTP POST requests based on the modern trolley bus positioning system [6], Shi Yansheng given under IPv6-based client-serverless Socket model and the use of UDP protocol for network communication method and to achieve the IPv6 protocol under the server-side and client data transmission between the information[7], P. Shrinivasan and others use the Socks protocol to complete the conversion of IPv6 and IPv4 data protocols[8], CS Li and others implement UPnP for IPv4 and IPv6[9], Others also use the http protocol for data exchange by the RESTful API[10].

2. Introduction

+ Corresponding author. Tel.: +86-10-58802190; fax: +86-10-58802190.
E-mail address: lixh@bnu.edu.cn.

TCP connections based on the IPv4 and IPv6 protocols are more commonly used methods of communication. In the original communication strategy, IPv4 was used to achieve the majority of communication. At present, most terminal communication is based on a single point with single-IP attach to network for communication. Within a local area network, the use of IP addresses in the private address for address allocation, When the number of terminals that need to access the public network increases, the terminal can't use the corresponding number of IPv4 addresses. In this case, IPv6 emerges as the times require. Compared with the 32-bit IPv4 address allocation, the biggest change in IPv6 is the expansion of the address space. 128-bit address space can ensure that any terminal has its own unique address, And IPv6 handles the security level for additional headers[11].

HTTP protocol, also known as hypertext transfer protocol, is mainly used for data transmission of Web applications, for C / S mode based on the terminal collecto, you can also use the HTTP protocol for data transmission. HTTP uses the client's initiative to achieve and server-side communication, the server passively accepts the request and returns the response data.

3. Implementation of Data Processing Methods

This article deals with data processing methods for the wireless sensor network terminal equipment to collect data using a variety of data transmission methods to be tested. Through experiments to verify the reliability of data communications and transmission efficiency with the using of IPv4, IPv6 and HTTP protocol.

3.1. IPv4 communication

In the early stage of project development, the client and server communication are based on the IPv4 protocol TCP connection, the terminal through the GPRS network to achieve with the server-side communication. The data message sent by the client encapsulates the data according to the agreed data format. The format of the current message data is as shown in Table 1, for different acquisition factors, data is arranged according to whether the corresponding data collector has data, and the collected data format is filled after the flag bits. In front of the data identification bits and data analysis part of the agreement, each one represents a different data sensor. In order to save the stored message length, the collected data will be filled in after the flag, otherwise it will not be filled. After the data was collected, analysing the data signs and the corresponding data to get the corresponding real data.

Table 1: Data transfer efficiency

Flag	…	Flag	Data	…	Data	DeviceId	Time
0	…	1	xxx	…	xxx	xxx	xxx

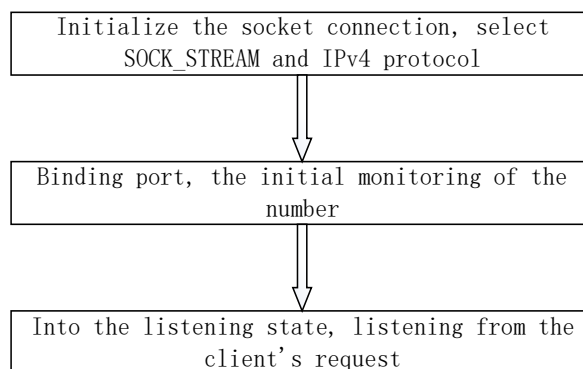Server-side TCP protocol based on the establishment of TCP monitoring process shown in Figure 1:

Fig. 1: TCP monitoring process based on IPv4 protocol

3.2. IPv6 communication

The IPv6 communication method is similar to that of IPv4. The difference is that the header structure of IPv6 packets is simpler than the header of IPv4 packets and is managed before the data for non-essential extension headers. Compared with the IPv4 protocol, packet processing efficiency is improved.

3.3. HTTP communication

TCP protocol is connection-oriented, and HTTP protocol is not connected：That is, each communication will be disconnected after the completion of such authentication defects. In response to this defect, identity authentication can be implemented through a token-ring based manner. Each communication carries its own identity instruction, and the server side performs identity verification by verifying the identity. HTTP communication process shown in Figure 2:

```
┌─────────────────────────────────────────┐
│      Server-side start web listener      │
└─────────────────────────────────────────┘
                     │
                     ▼
┌─────────────────────────────────────────┐
│   The client initiates an HTTP request   │◄──────┐
│                connection                │       │
└─────────────────────────────────────────┘       │
                     │                             │
                     ▼                             │
┌─────────────────────────────────────────┐       │
│  The server gateway program verifies the │       │
│   connection availability and redirects  │       │
│   the connection that needs to be        │       │
│              redirected                  │       │
└─────────────────────────────────────────┘       │
                     │                             │
                     ▼                             │
┌─────────────────────────────────────────┐       │
│  The server control program processes    │  Continue sending the request
│          HTTP request information        │       │
└─────────────────────────────────────────┘       │
                     │                             │
                     ▼                             │
┌─────────────────────────────────────────┐       │
│  The server will generate the information│       │
│           returned to the client         │       │
└─────────────────────────────────────────┘       │
                     │                             │
                     ▼                             │
┌─────────────────────────────────────────┐       │
│  The client receives the return message  │───────┘
│              and disconnects              │
└─────────────────────────────────────────┘
```
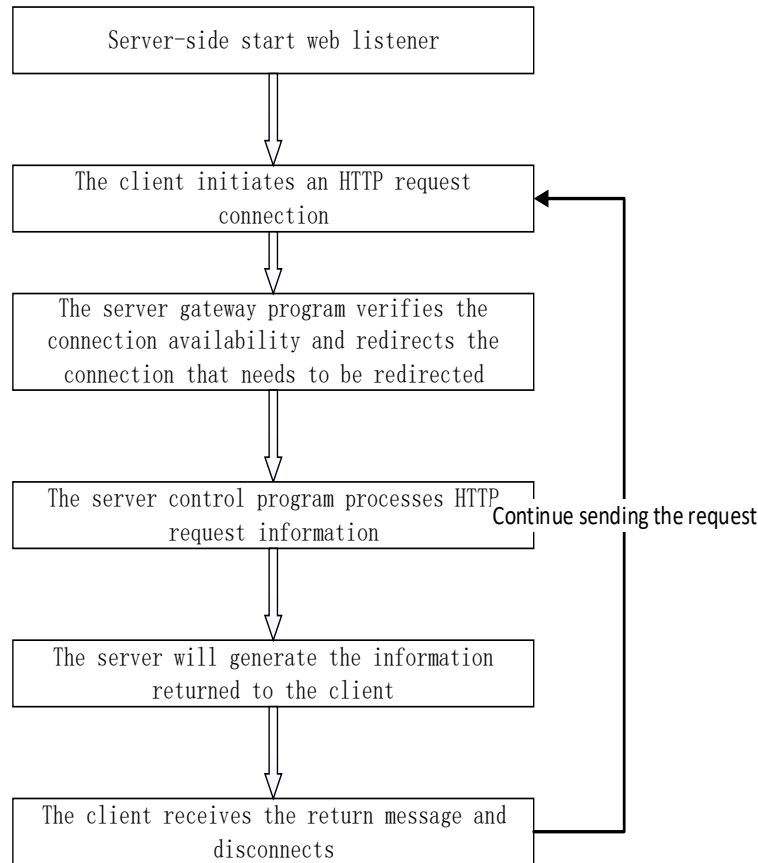
Fig. 2: Data communication process based on HTTP protocol

The advantage of HTTP connection over TCP connection is that server-side pressure is reduced. After each communication connection is completed, it is disconnected and the occupied resources are released. The disadvantage is that you can't maintain the previous connection information, and need to achieve other means of information retention and certification.

4. Experiment

In the experiment, we first set up the TCP listening service based on IPv6 and IPv4 protocols, and then obtain the corresponding packet length and data for the same packet upload. The efficiency of packet acquisition is compared to analysis the advantages and disadvantages of the two transmission protocols. Then the RESTful API for HTTP protocol is used to verify the stability of data transmission under the actual architecture.

4.1. Data transfer rate experiment

Because the collection network of the terminal collector does not adopt the dual-stack design, the experimental equipment is divided into two parts with the IPv4 and IPv6 protocols respectively for data upload. The data packets are uploaded through the communication between the TCP and the server. This article uses seven acquisition terminal equipments, and constantly send a connection request to the server to achieve data upload, the server for two different transport protocols, the establishment of the corresponding server-side receiving procedures. Through the stress test, get the actual server to accept the program data transmission efficiency as shown in Table 2:

Table 2: Data transfer efficiency

Protocol	The total length of the message	Total data length	Data transfer efficiency
IPv4	62	8	12.9%
IPv4	64	10	15.6%
IPv4	534	480	89.9%
IPv6	82	8	9.8%
IPv6	84	10	11.9%
IPv6	554	480	86.7%

Through the efficiency comparison, for the same message data, IPv4 data transmission efficiency is relatively high, because the IPv6 protocol's optional header data is moved to the optional data, but for the basic header, the increase in address space decreases the packet transmission rate. At the same time the expansion of address space compared to IPv4 to ensure that each terminal can be directly connected with the server without the need for terminal acquisition equipment for ad hoc network, reducing terminal consumption. Although the relative IPv4 data transmission efficiency of IPv6 has been reduced, when the amount of entity data is much larger than that of the message header, the difference in efficiency can be neglected, Due to the huge IPv6 address space, it is more applicable in IoT(Internet of Things) communications.

4.2. Response rate experiment of TCP connections based on IPv4 and IPv6 protocols

A TCP listening server based on IPv4 and IPv6 was established. The server 2008 r2 server used in this project was stress tested by the pressure measuring tool to test the success rate of data response. And we found that the server-based server to establish a link to TCP is mainly limited by the number of available random ports and server memory, when the remaining available ports more, about more than 3900 links can be established for the data received in the buffer for processing When the need for the corresponding number of memory processing. Using TCP test data as shown in Table 3:

Table 3: Data response rate

The number of requests	Server response number
IPv4	62
IPv4	64
IPv4	534
IPv6	82
IPv6	84
IPv6	554

Through data comparison, based on IPv4 and IPv6 TCP communication, the data stability is not high, for a specific server, high volume request connection may fail to connect because of lack of server resources.

4.3. HTTP data transfer experiment

Data transmission based on the HTTP protocol adopts the internationally accepted RESTful API to implement data interaction processing, and corresponding data operations are encapsulated under the same processing interface for different data models.

For example, for the registration of the terminal device, the client access API interface is as follows:

POST /api/device/register HTTP/1.1

Entity data

After receiving the client data, the client performs data analysis and returns the header of the data packet as follows:

```
HTTP/1.1 200 OK
Date: Tue, 30 Jan 2018 01:49:12 GMT
Last-Modified: Tue, 30 Jan 2018 01:48:53 GMT
Content-Type: text/plain
Transfer-Encoding: chunked
```

This paper verifies the experiment based on the use of HTTP communication while the 573 nodes was installed. The terminal nodes use the carrier communication to send the data to the region's relay gateway, then the gateway sends the data to server, the structure shown in Figure 3 Show:
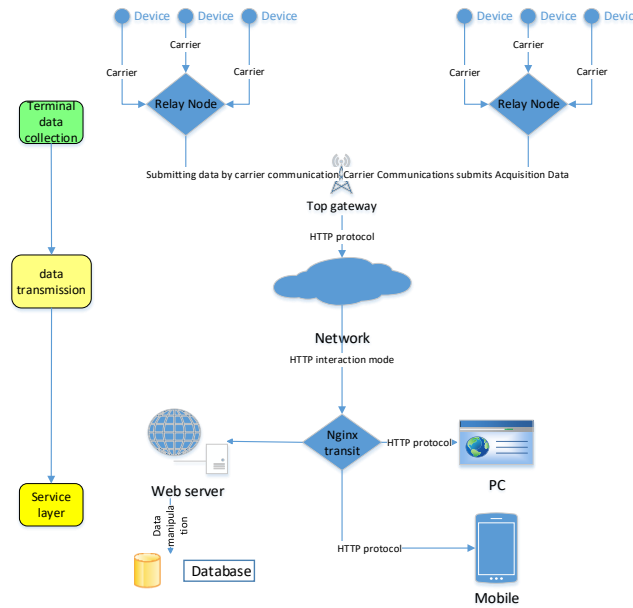

Fig. 3: Communication architecture based on HTTP protocol

Through the actual project verification, the current terminal-based collection device based on HTTP protocol uses the connection or disconnection way to realize the data interaction, which can transfer the data to the server in real time. The actual project test has not found the data loss service, and at the same time, and the secondary data was reported by data response.

5. Conclusion

Based on the analysis and actual use of multiple data transmission protocols, this paper compares the TCP connection based on IPV4 protocol and IPV6 protocol and the stateless transmission protocol based on HTTP. Experimental result shows that the communication protocol based on TCP connection can maintain two-way communication connection between the client and the server, but it consumes too much for the server and the data stability needs to be improved. Compared with IPv6 protocol, IPv4 transmits data slightly more efficiently but has a complex header and data analysis pressure is relatively large. The use of IPv6 wide address space can provide massive addresses, to achieve multi-terminal matching the corresponding address and then direct communication with the server. The stateless protocol Http consumes less for the server, but the disadvantage of Http is that it can't record the state of two-way communication. Therefore, IPv4-based TCP connections are more suitable for scenarios with higher server capabilities and fewer nodes and need to communicate bidirectionally. IPv6 can provide better results for applications that have massive nodes while requiring two-way communication; For the requirements of data stability, reduce server pressure, then the HTTP protocol superior. Therefore, there are different needs for a particular project, different transmission protocols can be used to achieve data transmission, in order to achieve the optimal transmission strategy in different scenarios.

6. Acknowledgment

7. References

[1]   Deering S, Hinden R. Internet Protocol, Version 6 (IPv6) Specification[M]. RFC Editor, 1995.

[2]   Chakravorty R, Vidales P, Subramanian K, et al. Practical experience with wireless networks integration using Mobile IPv6[J]. Acm Sigmobile Mobile Computing & Communications Review, 2003, 7(4):47-49.

[3]   Xiao Gelin. HTTP protocol technology analysis [J] .Jiangxi Communications Science and Technology, 2001(1):17-24.

[4]   Gao Song. Based on the HTTP protocol embedded remote monitoring system [D]. Beijing University of Technology, 2009.

[5]   Zun - min, Qu Da - yi, Chen Xiu - feng, et al.Traffic congestion information sharing system based on TCP / HTTP [J]. Journal of Chongqing Jiaotong University (Natural Science Edition), 2014, 33 (5): 107-110.

[6]   Chen Rongchao, Yang Hall. Modern tram locating system based on TCP Socket and HTTP POST [J]. Computer Applications and Software, 2016, 33(10):6-8.

[7]   Shi Yan-sheng, Liu Li-qiang. Realization of UDP communication based on IPv6 [J]. Computer Technology and Development, 2006, 16(5):191-193.

[8]   Patnaikuni P S R, Kulkarni R B. An architecture for "Web Of Things" using SOCKS protocol based IPv6/IPv4 gatewaying for heterogeneous communication[J]. Advances in Internet of Things, 2012, 2(1):8-12.k

[9]   Li C S, Huang Y M, Chao H C. UPnP IPv4/IPv6 bridge for home networking environment[J]. IEEE Transactions on Consumer Electronics, 2008, 54(4):1651-1655.

[10]   Leonard Richardson, Sam Ruby. Restful web services[J]. O'reilly Media Inc, 2007, 4(Sept):199 - 204.

[11]   Durdağı E, Buldu A. IPV4/IPV6 security and threat comparisons[J]. Procedia - Social and Behavioral Sciences, 2010, 2(2):5285-5291.