

使用paramiko实现网络设备 自动化巡检

学院：信息工程学院

教师：张迁

目录

1. 网络设备巡检概念
2. paramiko模块
3. 网络设备配置SSH服务
4. 实训4-使用python脚本实现自动化巡检

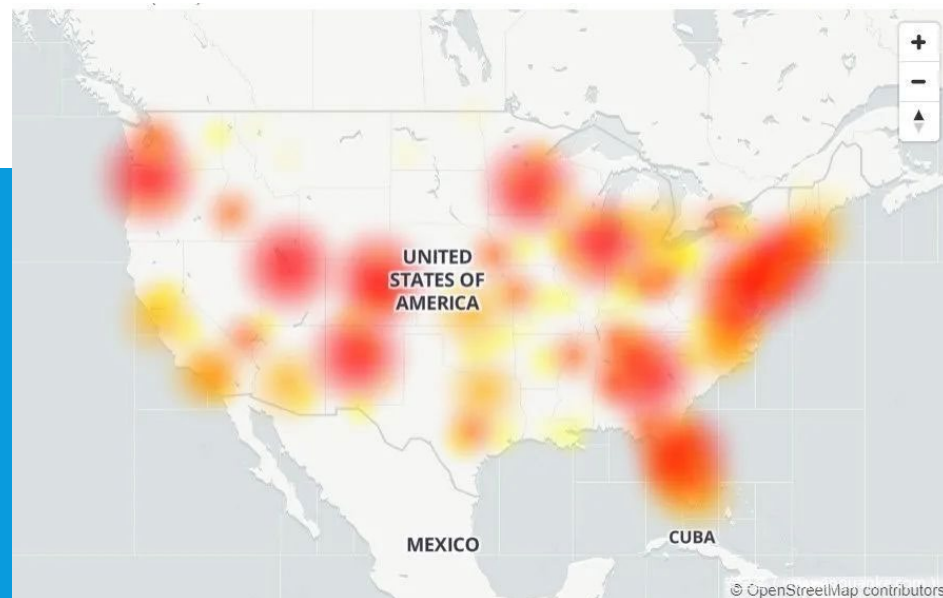
1.网络设备巡检概念

- 什么是网络设备巡检
- 网络设备巡检基本信息库
- 网络设备基本信息检查
- 设备运行检查
- 端口检查
- 业务检查

1.1 什么是网络设备巡检

- 设备稳定运行一方面依赖于完备的网络规划，另一方面依赖于日常的维护和监控。
- 网络巡检时指通过标准的方法和流程，定期对一定范围内的网络进行网元级的系统检查，内容包括现场数据采集、分析和报告生成等。

PS: 网元（Network Element）通常通常是通信网络中的一个物理或虚拟设备，它可以是硬件设备，如路由器、交换机、基站等；也可以是软件组件，如协议处理软件、应用服务等。



Priority Colo @PColoMaint · Aug 30, 2020
Replying to @PColoMaint

We are still monitoring the situation. Unsurprisingly Centurylink does not have the manpower to answer their phones during a global outage, and is providing incredibly brief updates VIA twitter (thus far), simply acknowledging they're aware of the issue.

Priority Colo @PColoMaint

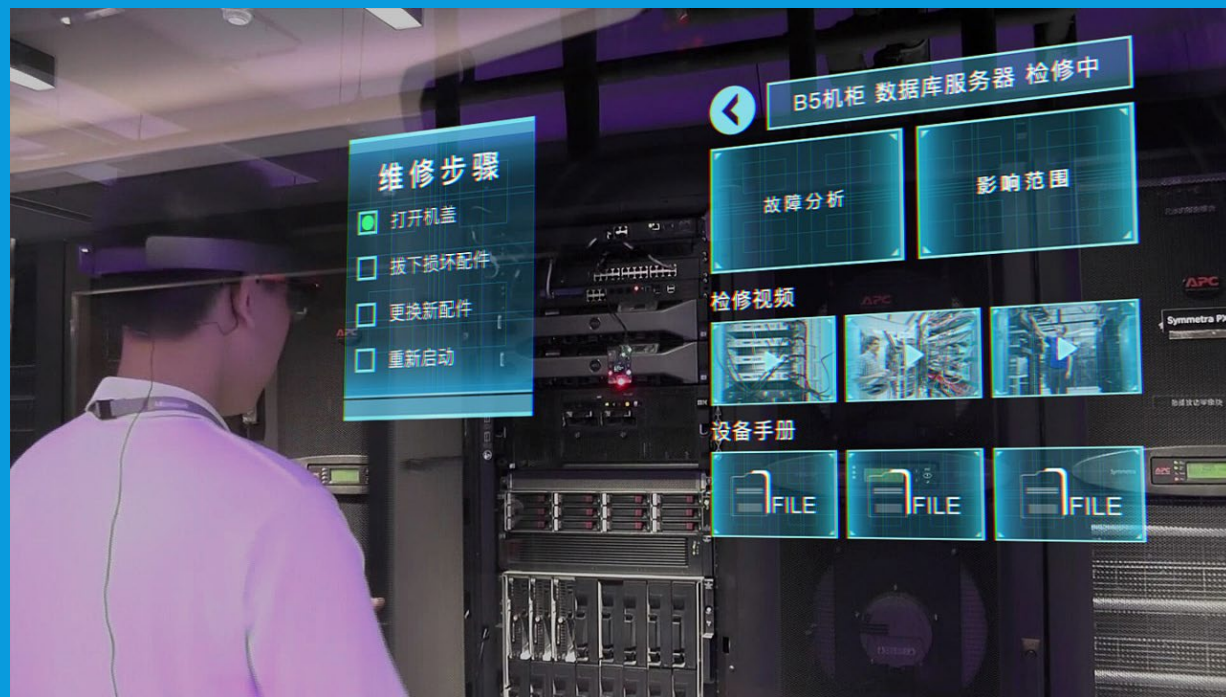
Centurylink appears to have withdrawn the invalid BGP routes as of ~10 minutes ago, and fixed their network woes. We are leaving our AS3356 session down for the time being as a precautionary measure, in case they have a relapse. We will continue to monitor throughout the day.

11:26 AM · Aug 30, 2020

1 See Priority Colo's other Tweets

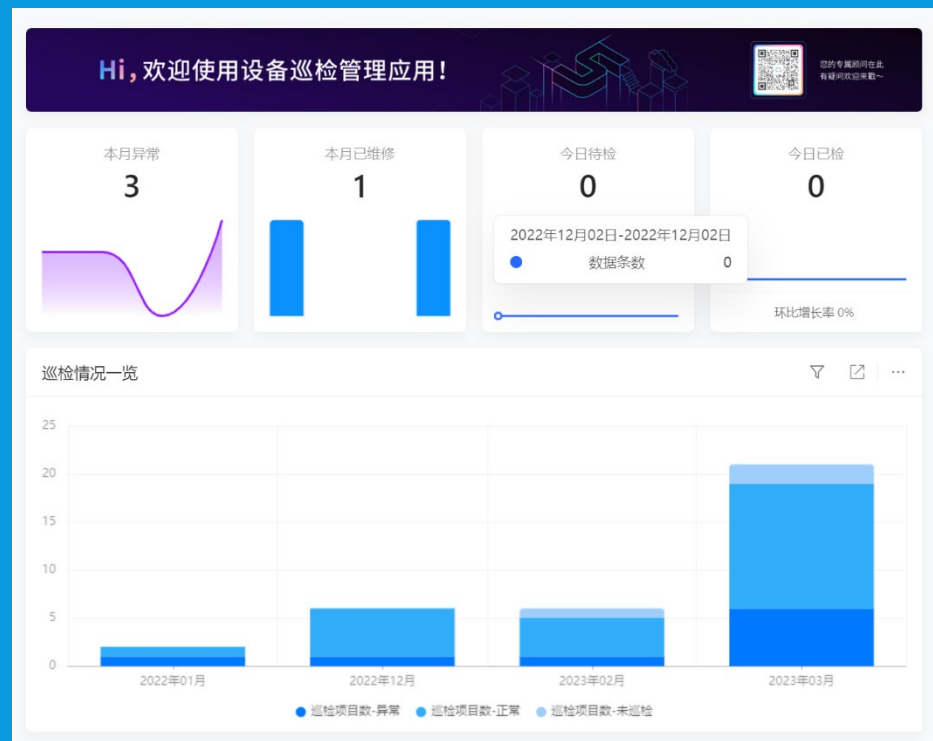
1.1 什么是网络设备巡检

- 通过定期网络巡检，可以及时发现网络中存在的隐患，并将其消灭在萌芽状态。一般中大型公司都需要对网络设备进行定期巡检。
- 当设备量比较大且巡检指标较多的时候，该项工作往往费时费力；如果完全采用人工巡检，还容易因人为因素出现失误。通过自动化工具或者编写Python脚本程序对网络设备进行自动化巡检，可以提高工作效率并且减少因人为因素出现的失误，从而提升网络服务质量，确保设备的正常运行。
- 通过自动化巡检收集相关的数据的结果进行对比，可以更加准确地了解网络系统的整体运行情况，并可以与手动数据采集的结果进行对比，确保数据采集和分析的合理性及可靠性。



1.2网络设备巡检基本信息库

由于网络系统的巡检是一个长期的、持续性的工作，因此需要对网络系统有一定的了解，建立一个基本信息库可以为后续巡检工作的数据对比提供基础和依据。在信息库建立后，需要保持数据更新、动态调整基本信息库的参考点。



1.2网络设备巡检基本信息库

基本信息库主要包括：

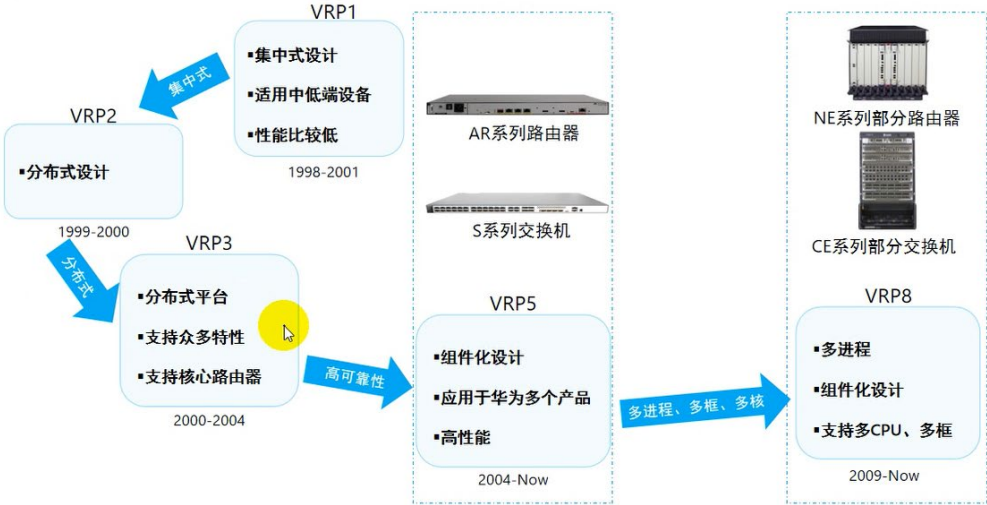
- 1、设备清单：包括设备名称、IP地址、位置、序列号等；
- 2、设备模块硬件配置：网络设备的模块种类和型号等；
- 3、设备软件版本：包括通用路由平台VRP软件版本、补丁和授权等；
- 4、设备使用情况，包括维修记录、使用时长等；
- 5、设备性能基准：包括CPU和内存使用率及设备端口流量等；
- 6、设备端口信息：包括端口密度、速率和计数器状态等。



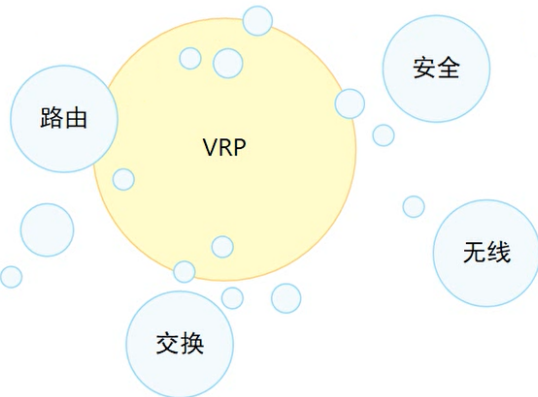
1.3网络设备基本信息检查



VRP的发展



什么是VRP?



- VRP是华为公司数据通信产品的通用操作系统平台，作为华为公司从低端到核心的全系列路由器、以太网交换机、业务网关等产品的软件核心引擎。
- VRP提供以下功能：
 - 实现统一的用户界面和管理界面
 - 实现控制平面功能，并定义转发平面接口规范
 - 实现各产品转发平面与VRP控制平面之间的交互
 - 屏蔽各产品链路层对于网络层的差异



VRP用户级别

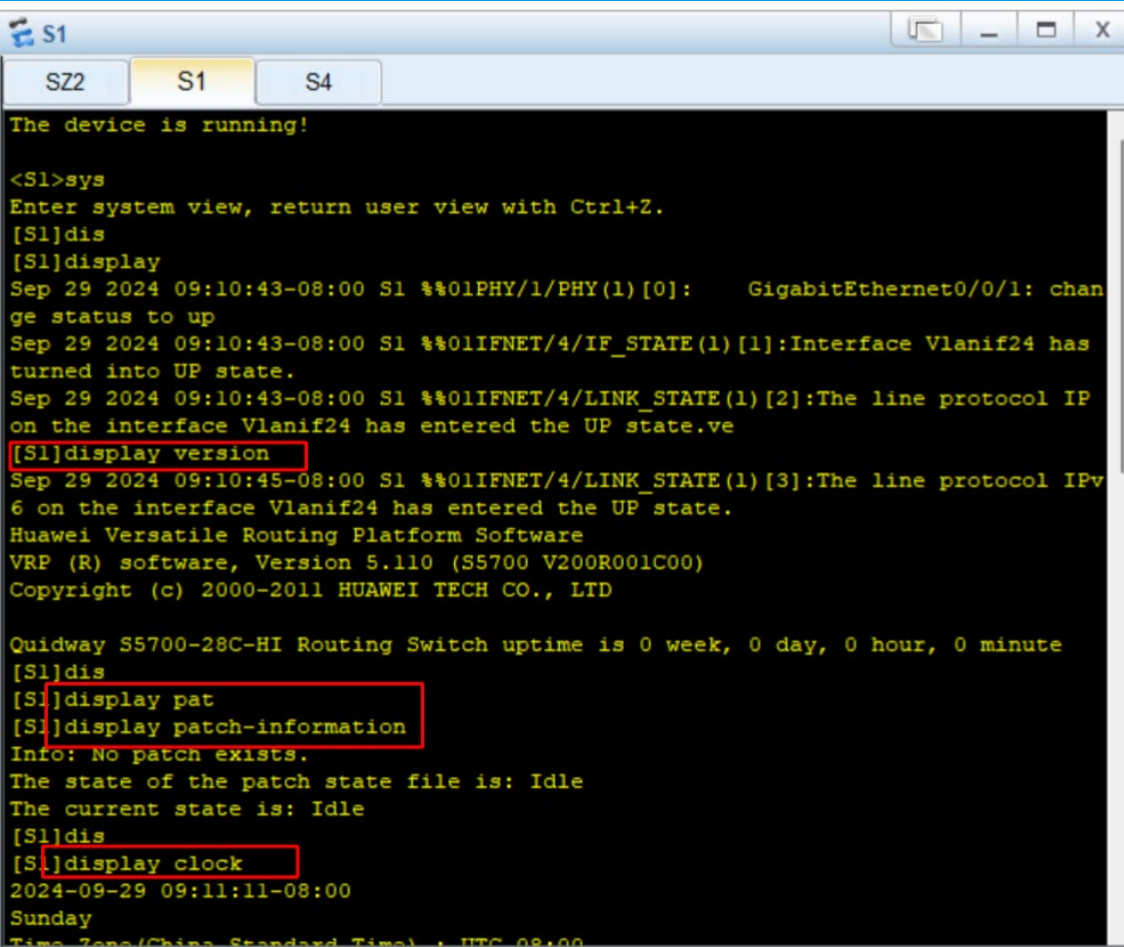
- VRP提供基本的权限控制，可以实现不同级别的用户能够执行不同级别的命令，用以限制不同用户对设备的操作。

用户等级	命令等级	名称	说明
0	0	参观级	可使用网络诊断工具命令（ping、tracert）、从本设备出发访问外部设备的命令（Telnet客户端命令）、部分display命令等。
1	0 and 1	监控级	用于系统维护，可使用display等命令。
2	0,1 and 2	配置级	可使用业务配置命令，包括路由、各个网络层次的命令，向用户提供直接网络服务。
3-15	0,1,2 and 3	管理级	可使用用于系统基本运行的命令，对业务提供支撑作用，包括文件系统、FTP、TFTP下载、命令级别设置命令以及用于业务故障诊断的debugging命令等。

1.3 网络设备基本信息检查

1. 网络设备基本信息检查主要是检查设备的基本信息，如设备软件版本信息、补丁信息和系统世界、Flash空间及配置信息。常见命令：

- 执行命令 `display version` 检查软件；
- 执行命令 `display patch-information` 检查补丁信息；
- 执行命令 `display clock` 检查系统时间；
- 执行命令 `dir flash` 检查flash空间等（此功能需要退出网络设备的视图模式）。



```
S1
SZ2  S1  S4

The device is running!

<S1>sys
Enter user view, return user view with Ctrl+Z.
[S1]dis
[S1]display
Sep 29 2024 09:10:43-08:00 S1 %%01PHY/1/PHY(1)[0]: GigabitEthernet0/0/1: change status to up
Sep 29 2024 09:10:43-08:00 S1 %%01IFNET/4/IF_STATE(1)[1]:Interface Vlanif24 has turned into UP state.
Sep 29 2024 09:10:43-08:00 S1 %%01IFNET/4/LINK_STATE(1)[2]:The line protocol IP on the interface Vlanif24 has entered the UP state.ve
[S1]display version
Sep 29 2024 09:10:45-08:00 S1 %%01IFNET/4/LINK_STATE(1)[3]:The line protocol IPv6 on the interface Vlanif24 has entered the UP state.
Huawei Versatile Routing Platform Software
VRP (R) software, Version 5.110 (S5700 V200R001C00)
Copyright (c) 2000-2011 HUAWEI TECH CO., LTD

Quidway S5700-28C-HI Routing Switch uptime is 0 week, 0 day, 0 hour, 0 minute
[S1]dis
[S1]display pat
[S1]display patch-information
Info: No patch exists.
The state of the patch state file is: Idle
The current state is: Idle
[S1]dis
[S1]display clock
2024-09-29 09:11:11-08:00
Sunday
Time Zone (China Standard Time) : UTC+08:00
```

1.4 设备运行检查

2. 设备运行检查主要是检查设备的运行情况，如CPU和内存使用率、设备端口和日志信息。常见命令：

- 执行命令 `display device` 检查设备的部件类型和状态信息；
- 执行命令 `display cpu-usage` 检查CPU利用率；
- 执行命令 `display memory-usage` 检查内存使用率；
- 执行命令 `display logbuffer summary` 检查设备日志信息等。

```
S1
SZ2  S1  S4
[S1]
[S1]
[S1]
[S1]display device
S5700-28C-HI's Device status:
Slot  Sub Type      Online  Power   Register  Status  Role
-----
0      -   5728C      Present PowerOn  Registered Normal  Master
[S1]dis
[S1]display cp
[S1]display cpu-
[S1]display cpu-defend
[S1]display cpu-usage
[S1]display cpu-defend
[S1]display cpu-usage
CPU Usage Stat. Cycle: 60 (Second)
CPU Usage           : 1% Max: 100%
CPU Usage Stat. Time : 2024-09-29 09:57:11
CPU utilization for five seconds: 1%: one minute: 1%: five minutes: 1%.

TaskName           CPU  Runtime(CPU Tick High/Tick Low) Task Explanation
BOX                 0%    0/ be3d6ce          BOX Output
_TIL                0%    0/          0      Infinite loop event task
VCLK                0%    0/10ec2eb7
TICK                0%    0/15b55130
co0                 0%    0/          0      co0 Line user's task
U0                  0%    0/          0      U0 user command process
task
RTMR                0%    0/ 3f38216          RTMR
IPCR                0%    0/          0      IPCR
VPR                 0%    0/          0      VPR
```

```
<S1>display memory-usage
Memory utilization statistics at 2024-09-29 09:58:29-08:00
System Total Memory Is: 171493452 bytes
Total Memory Used Is: 124396612 bytes
Memory Using Percentage Is: 72%
<S1>
```

1.5 端口检查

3. 端口检查主要是检查设备的端口信息，如端口篇日志、端口状态等是否正确。
常见命令：

- 执行命令 `display interface` 检查当前运行状态和接口统计信息；
- 执行命令 `display current-configuration interface` 检查端口配置；
- 执行命令 `display interface brief` 检查端口 Up/Down 状态。

```
[S1]display in
[S1]display increment-command
[S1]display increment-synchronization-result
[S1]display interface
Eth-Trunk12 current state : UP
Line protocol current state : UP
Description:
Switch Port, PVID : 1, Hash arithmetic : According to SIP-XOR-DIP,Maximal BW:
2G, Current BW: 2G, The Maximum Frame Length is 9216
IP Sending Frames' Format is PKTFMT_ETHNT_2, Hardware address is 4c1f-cc32-6569
Current system time: 2024-09-29 10:04:18-08:00
    Input bandwidth utilization : 0%
    Output bandwidth utilization : 0%
-----
PortName                Status    Weight
-----
GigabitEthernet0/0/23    UP        1
GigabitEthernet0/0/24    UP        1
-----

[S1]display interface br
[S1]display interface brief
PHY: Physical
*down: administratively down
(l): loopback
(s): spoofing
(b): BFD down
(e): ETHOAM down
(dl): DLDP down
(d): Dampening Suppressed
InUti/OutUti: input utility/output utility
Interface                PHY    Protocol  InUti  OutUti    inErrors  outErrors
Eth-Trunk12              up     up        0%    0%        0         0
  GigabitEthernet0/0/23   up     up        0%    0%        0         0
  GigabitEthernet0/0/24   up     up        0%    0%        0         0
GigabitEthernet0/0/1     up     up        0%    0%        0         0
GigabitEthernet0/0/2     down   down      0%    0%        0         0
GigabitEthernet0/0/3     down   down      0%    0%        0         0
GigabitEthernet0/0/4     down   down      0%    0%        0         0
GigabitEthernet0/0/5     down   down      0%    0%        0         0
```

1.6 业务检查

4. 业务检查主要是检查设备运行的业务是否正确。常见命令：

- 执行命令 `display dhcp snooping user-bind all` 检查 DHCP Snooping 绑定表；
- 执行命令 `display mac-address` 检查 MAC 地址表信息；
- 执行命令 `display ip routing-table` 检查路由表信息。

```
[S1]display ip ro
[S1]display ip routing-table
Route Flags: R - relay, D - download to fib
-----
Routing Tables: Public
      Destinations : 19          Routes : 19

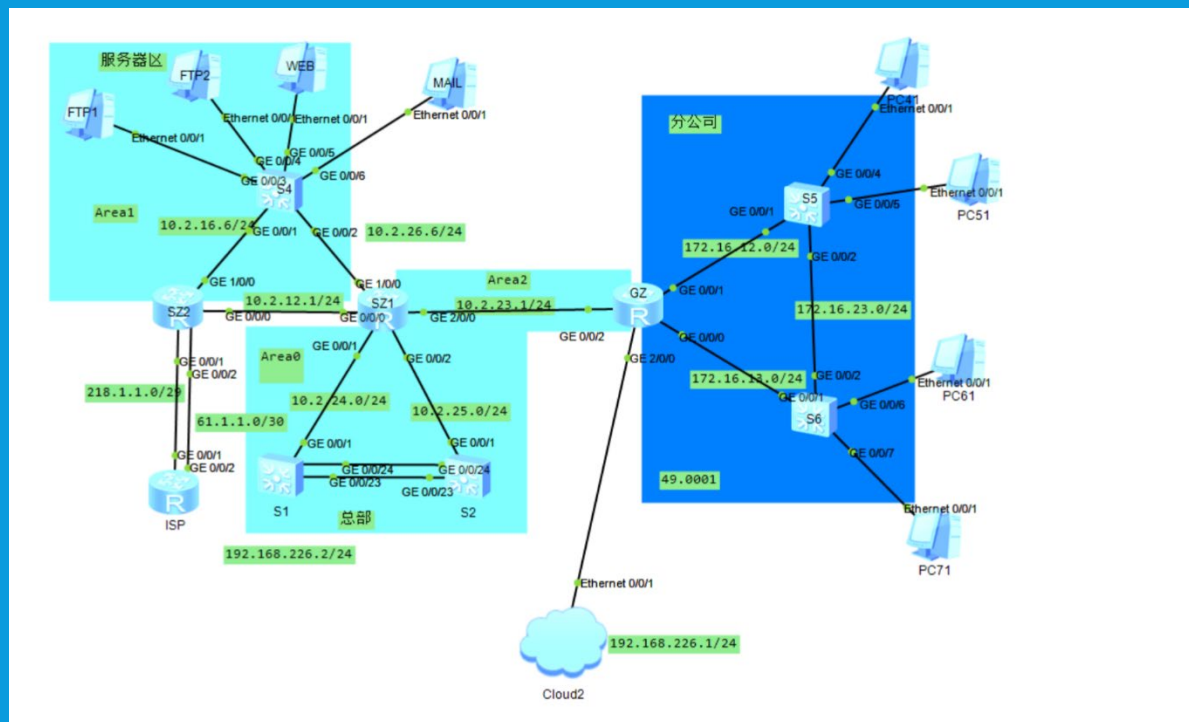
Destination/Mask    Proto    Pre  Cost           Flags NextHop          Interface
-----
0.0.0.0/0           O_ASE    150   1             D    10.2.24.1          Vlanif24
10.1.4.0/24         Direct   0     0             D    10.1.4.252         Vlanif4
10.1.4.252/32       Direct   0     0             D    127.0.0.1          Vlanif4
10.1.5.0/24         Direct   0     0             D    10.1.5.252         Vlanif5
10.1.5.252/32       Direct   0     0             D    127.0.0.1          Vlanif5
10.1.6.0/24         Direct   0     0             D    10.1.6.252         Vlanif6
10.1.6.252/32       Direct   0     0             D    127.0.0.1          Vlanif6
10.1.7.0/24         Direct   0     0             D    10.1.7.252         Vlanif7
10.1.7.252/32       Direct   0     0             D    127.0.0.1          Vlanif7
10.2.12.0/24        OSPF     10     2             D    10.2.24.1          Vlanif24
10.2.16.0/24        OSPF     10     3             D    10.2.24.1          Vlanif24
10.2.23.0/24        OSPF     10     2             D    10.2.24.1          Vlanif24
10.2.24.0/24        Direct   0     0             D    10.2.24.4          Vlanif24
10.2.24.4/32        Direct   0     0             D    127.0.0.1          Vlanif24
10.2.25.0/24        OSPF     10     2             D    10.2.24.1          Vlanif24
10.2.26.0/24        OSPF     10     2             D    10.2.24.1          Vlanif24
10.3.1.0/24         OSPF     10     3             D    10.2.24.1          Vlanif24
127.0.0.0/8         Direct   0     0             D    127.0.0.1          InLoopBack0
127.0.0.1/32        Direct   0     0             D    127.0.0.1          InLoopBack0

[S1]dis
[S1]display dh
[S1]display dhcp sn
[S1]display dhcp snooping us
[S1]display dhcp snooping user-bind a
[S1]display dhcp snooping user-bind all
Info: The number of dhcp snooping bind-table is zero.
[S1]
```

实训4-（一）：对下列网络拓扑图中设备GZ进行手动巡检

【任务目标】基于指导教师给的网络拓扑图Ensp文件，对GZ设备进行巡检：

- 1.查看GZ的软件版本、检查当前配置、检查Flash空间。
- 2.查看GZ的内存使用率和日志信息。
- 3.查看GZ的端口状态、端口配置信息。
- 4.查看GZ的DHCP绑定表和路由表信息。



2. paramiko 模块

- 什么是paramiko
- paramiko模块的安装
- paramiko模块的基本组件
- SSHClient的常用方法
- SFTPClient类的常用方法

2.1 什么是paramiko

paramiko是一个通过Python实现SSH协议的模块，它提供了SSH客户端和服务器的功能，可以让Python程序通过SSH协议连接到远程主机或网络设备，实现执行命令和传输文件等基本操作。

Paramiko是Python实现SSHv2协议的模块，支持口令认证和公钥认证两种方式，支持SSH隧道、代理和密钥。



2.2 paramiko模块的安装

Paramiko模块不是Python的标准模块，如之前Python编程基础课程所述，需要安装后才可以导入该模块进行使用。

Paramiko安装命令：

pip install paramiko

PS：记得在Anaconda Prompt中激活虚拟环境ensp_env后，再进行安装哦！

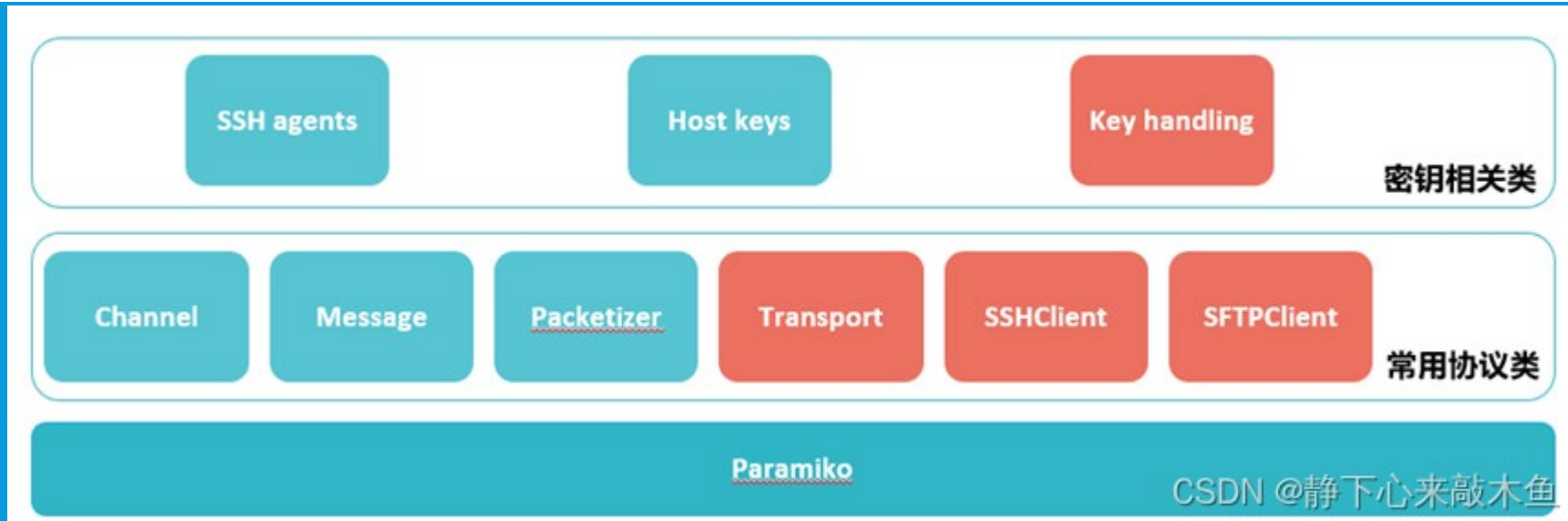
```
(base) C:\Users\USSTz>conda activate ensp_env
(ensp_env) C:\Users\USSTz>pip install paramiko
Requirement already satisfied: paramiko in c:\users\usstz\anaconda3\envs\ensp_
Requirement already satisfied: bcrypt>=3.2 in c:\users\usstz\anaconda3\envs\en
3.2.0)
Requirement already satisfied: cryptography>=3.3 in c:\users\usstz\anaconda3\en
iko) (42.0.5)
Requirement already satisfied: pynacl>=1.5 in c:\users\usstz\anaconda3\envs\en
1.5.0)
Requirement already satisfied: cffi>=1.1 in c:\users\usstz\anaconda3\envs\ensp
paramiko) (1.16.0)
Requirement already satisfied: six>=1.4.1 in c:\users\usstz\anaconda3\envs\ens
>paramiko) (1.16.0)
Requirement already satisfied: pycparser in c:\users\usstz\anaconda3\envs\ensp
rypt>=3.2->paramiko) (2.21)
(ensp_env) C:\Users\USSTz>
```

2.2 paramiko模块的安装

安装后，在Anaconda Prompt测试 paramiko，可以看出 paramiko 的版本为2.7.2

```
C:\Users\Administrator>python
Python 3.9.6 (tags/v3.9.6:db3ff76, Jun 28 2021, 15:26:21) [MSC v.1929 64 bit (AMD64)] on
win32
Type "help", "copyright", "credits" or "license" for more information.
>>>
>>> import paramiko
>>> paramiko.__version__
'2.7.2'
>>>
>>> exit()
```

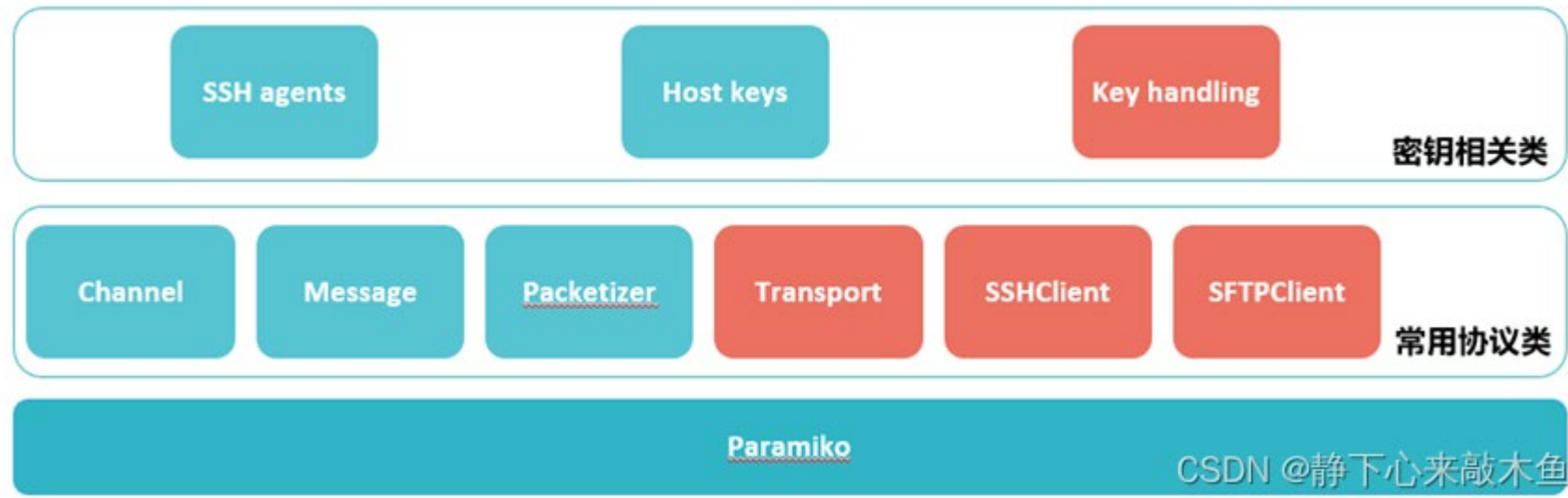
2.3 paramiko模块的基本组件



- Channel类：该类用于创建在SSH Transport上的安全通道。
- Message类：SSH Message是字节流。该类对字符串、整数等进行编码。
- Packetizer类：数据包处理类。
- Transport类：该类用于在现有套接字或类套接字对象上创建一个Transport会话对象。
- SFTPClient类：该类通过一个打开的SSH Transport会话创建SFTP会话通道并执行远程文件操作。
- SSHClient类：该类是与SSH服务器会话的高级表示，集成了Transport,Channel和SFTPClient类。

Paramiko模块常用的两个类为SSHClient和SFTPClient类，分别提供SSH和SFTP功能。

2.3 paramiko模块的基本组件



密钥相关类主要有以下组件：

- SSH Agent类：该类用于SSH代理。
- Host keys类：该类与OpenSSH `known_hosts`文件相关，用于创建一个host keys对象。
- Key handling类：该类用于创建对应密钥类型的实例，如RSA密钥，DSS（DSA）密钥。

OpenSSH是SSH协议的免费开源实现。OpenSSH提供了服务端后台程序和客户端工具。所有的Linux操作系统均集成了OpenSSH。OpenSSH把用户访问过每个计算机的公钥都记录在`~/.ssh/known_hosts`。当下次访问相同计算机时，OpenSSH会核对公钥。如果公钥不同，OpenSSH会发出警告，避免用户受到中间人攻击等。

2.4 SSHClient的常用方法

SSHClient类是对SSH会话的封装，它封装了Transport类、Channel类和SFTPClient类来进行会话通道的建立及鉴权验证，通常用于执行远程命令。

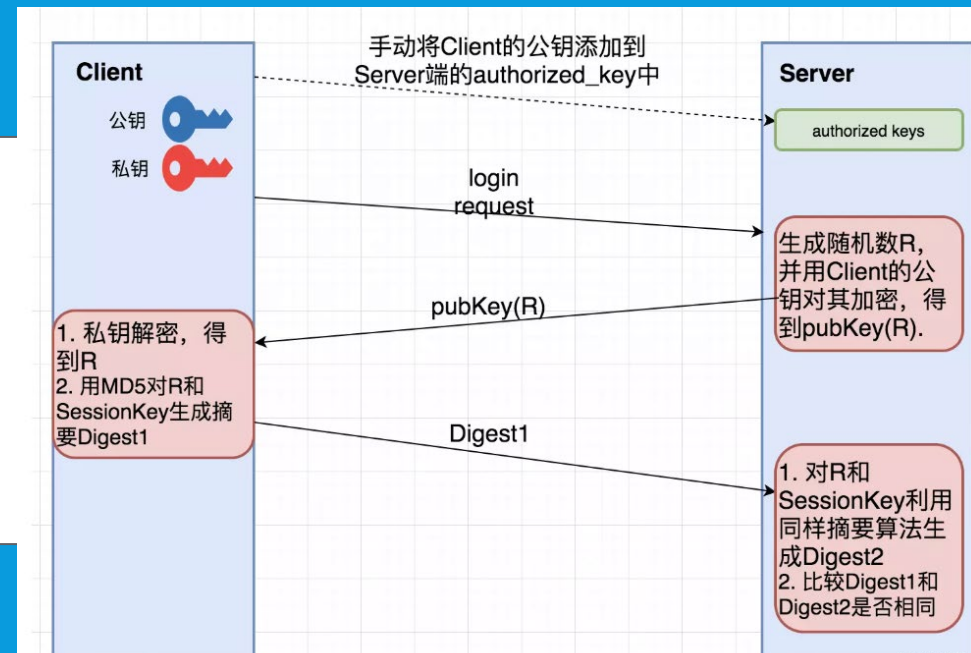
方法	功能
<code>connect()</code>	实现远程服务器的连接与验证
<code>set_missing_host_key_policy()</code>	设置连接到没有已知主机密钥的服务器时使用的策略
<code>load_system_host_keys()</code>	从系统文件中加载主机密钥
<code>invoke_shell()</code>	在远程服务器上启动交互式Shell会话
<code>exec_command()</code>	在远程服务器执行Linux命令
<code>open_sftp()</code>	在一个会话链接中创建SFTP通道
<code>close()</code>	关闭建立的连接

2.4 SSHClient的常用方法

1. `connect()`用于实现远程服务器的连接与验证，常用参数如下：

`connect(hostname, port=22, username=None, password=None, pkey=None, key_filename=None, timeout=None, allow_agent=True, look_for_keys=True, compress=False, sock=None, gss_auth=False, gss_kex=False, gss_deleg_creds=True, gss_host=None, banner_timeout=None, auth_timeout=None, gss_trust_dns=True, passphrase=None, disabled_algorithms=None)`

Hostname: 连接的目标主机; port: SSH_PORT 指定端口
username: 验证的用户名; password: 验证的用户密码
pkey: 私钥方式用于身份验证; key_filename: 一个文件名或文件列表, 指定私钥文件
timeout: 可选的tcp连接超时时间; compress: 是否打开压缩
allow_agent: 是否允许连接到ssh代理, 默认为True 允许
look_for_keys: 是否在~/.ssh中搜索私钥文件, 默认为True 允许



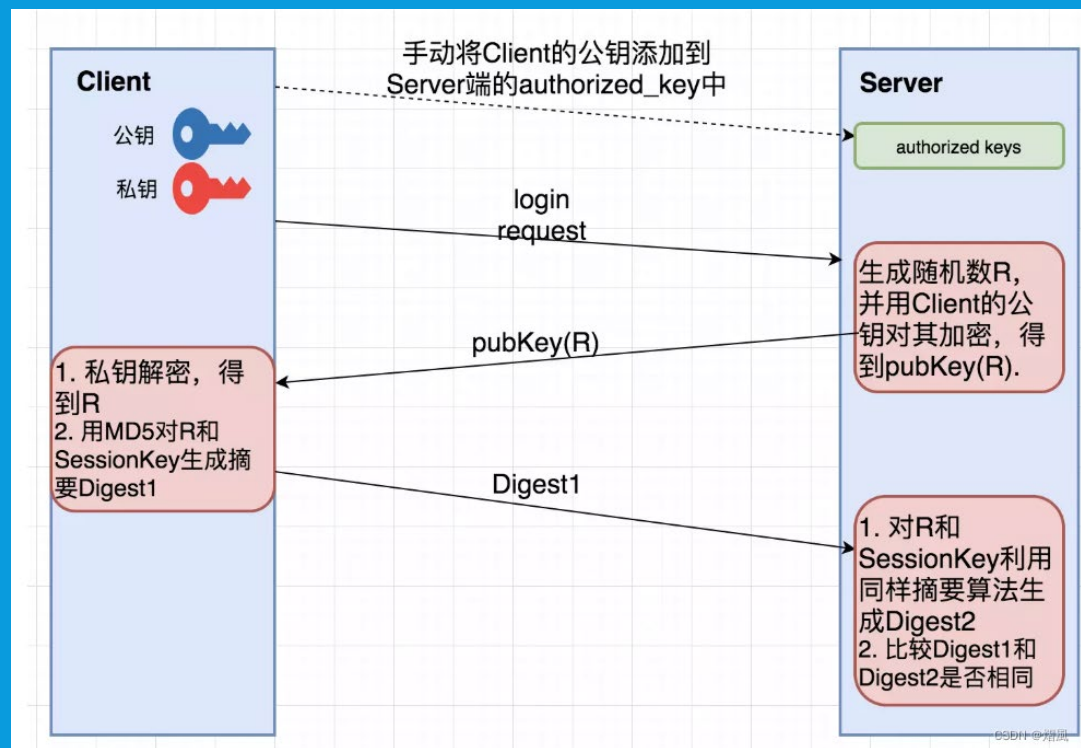
2.4 SSHClient的常用方法

2. `set_missing_host_key_policy()`: 设置连接的远程主机没有本地主机密钥或HostKeys对象时的策略，目前支持三种：

AutoAddPolicy 自动添加主机名及主机密钥到本地HostKeys对象，不依赖load_system_host_key的配置。即新建立ssh连接时不需要再输入yes或no进行确认

WarningPolicy 用于记录一个未知的主机密钥的python警告。并接受，功能上和AutoAddPolicy类似，但是会提示是新连接

RejectPolicy 自动拒绝未知的主机名和密钥，依赖load_system_host_key的配置。此为默认选项



2.4 SSHClient的常用方法

3. `load_system_host_keys()` 该方法用于从系统文件加载主机密钥，如果没有参数，那么尝试从用户本地的known hosts文件中读取密钥信息。

4. `invoke_shell()` 该方法用于基于SSH会话连接，启动一个交互式Shell会话：

```
cli = client.invoke_shell()
```

5. `exec_command()` 在远程服务器执行Linux命令的方法：

```
stdin, stdout, stderr = client.exec_comand("ls -l")
```

6. `open_sftp()` 用于在一个连接中创建SFTP会话：

```
sftp = client.open_sftp()
```

load_system_host_keys(*filename=None*)

Load host keys from a system (read-only) file. Host keys read with this method will not be saved back by **save_host_keys**.

This method can be called multiple times. Each new set of host keys will be merged with the existing set (new replacing old if there are conflicts).

If `filename` is left as `None`, an attempt will be made to read keys from the user's local "known hosts" file, as used by OpenSSH, and no exception will be raised if the file can't be read. This is probably only useful on posix.

Parameters: `filename` (*str*) – the filename to read, or `None`

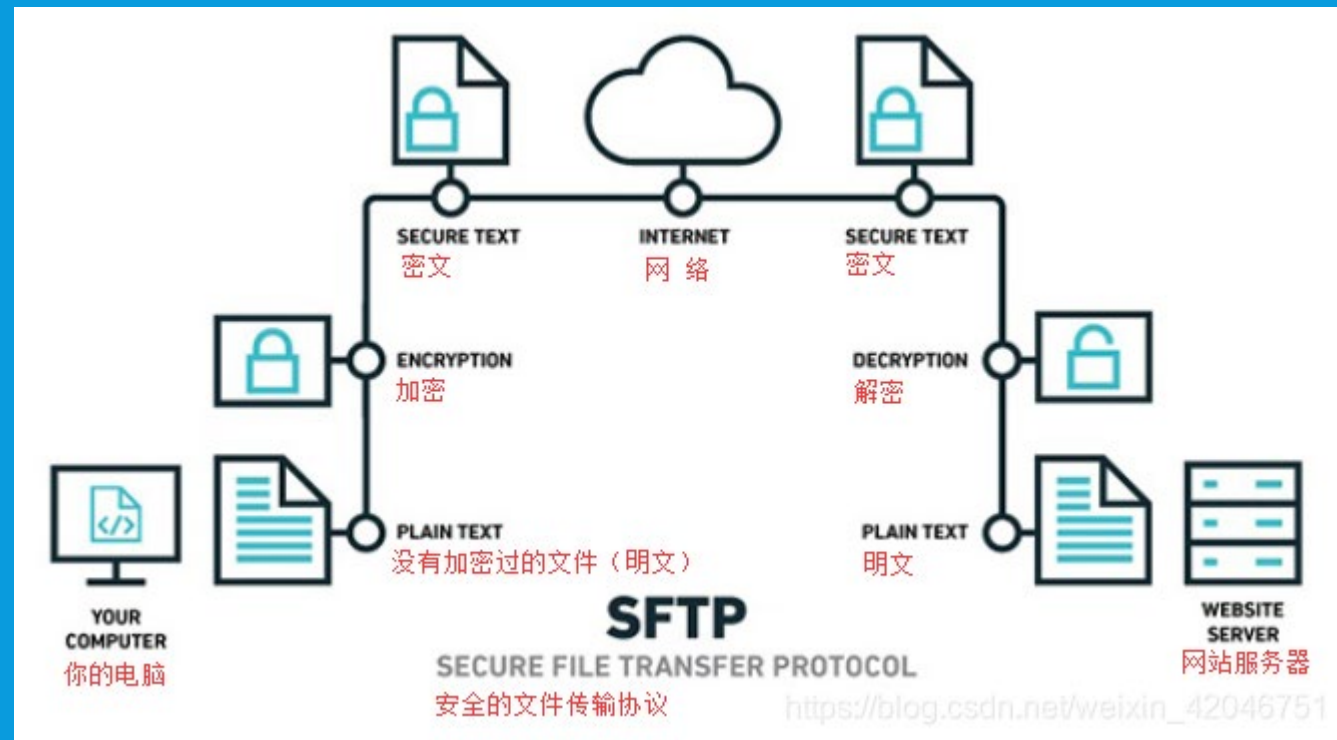
Raises: `IOError` – if a filename was provided and the file could not be read

利用SSHClient对象的`open_sftp()`方法，可以直接返回一个基于当前连接的sftp对象，可以进行文件的上传等操作。如

```
sftp = client.open_sftp()
sftp.put('test.txt','text.txt')
```

2.5 SFTPClient的常用方法

SSH文件传送协议（SFTP）是一个安全的文件传输协议，建立在SSH协议的基础上，SFTP不仅提供了文件传送协议FTP的所有功能，而且安全性和可靠性更高。在作为SFTP服务器的网络设备上使能SFTP服务器功能后，客户端可以通过密钥验证等方法登录SFTP服务器实现文件传输。



2.5 SFTPClient的常用方法

SFTPClient类通过一个打开的SSH Transport会话通道创建SFTP会话连接并支持远程文件操作。常用方法如下：

方法	功能
<code>from_transport()</code>	通过
<code>set_missing_host_key_policy()</code>	设置连接到没有已知主机密钥的服务器时使用的策略
<code>load_system_host_keys()</code>	从系统文件中加载主机密钥
<code>invoke_shell()</code>	在远程服务器上启动交互式Shell会话
<code>exec_command()</code>	在远程服务器执行Linux命令
<code>open_sftp()</code>	在一个会话链接中创建SFTP通道
<code>close()</code>	关闭建立的连接

2.5 SFTPClient的常用方法

- SFTPClient类作为一个sftp的客户端对象，根据SSH传输协议的sftp会话，实现远程文件操作，如上传、下载、权限、状态。常用的方法：

- **from_transport():** 从开启的Transport通道创建一个SFTP客户端通道。

常用到的参数：

T： 一个认证过的开启的Transport会话；

window_size： 可选参数，SFTP会话窗口大小

max_packet_size： 可选参数，SFTP会话最大数据包大小

2.5 SFTPClient的 常用方法

- `get()`: 将远程文件 (`remotepath`) 从SFTP服务器复制到本地主机的指定路径中 (`localpath`) , 操作引发的任何异常都将被传递。
- `put()`: 将本地文件(`localpath`)从本地主机复制到SFTP服务器的指定路径中(`remotepath`) , 操作引发的任何异常都将被传递。
- `mkdir()` 在服务器上创建目录
- `remove()` 在服务器上删除目录
- `rename()` 在服务器上重命名目录
- `stat()` 查看服务器文件状态
- `listdir()` 列出服务器目录下的文件

2.5 SFTPClient的 常用方法

- Key handling类用于创建对应密钥类型的实例，如RSA密钥，DSS（DSA）密钥。这个类包含了密钥的读取，写入等相关方法。常用方法：

- `RSAKey.from_private_key_file(filename)`: 从文件读取RSA私钥来创建密钥对象
- `DSSKey.from_private_key_file(filename)`: 从文件读取DSS私钥来创建密钥对象

实训4-（二）：安装paramiko并验证版本

【任务目标】基于教师的PPT文件，完成paramiko安装及使用：

- 1.通过Anaconda Prompt在虚拟环境ensp_py下安装paramiko包，并验证其版本。
- 2.假设远程主机ip为“192.168.1.250”，用户名为“python”，密码为“Huawei12#\$”，如何通过paramiko模块的SSHClient类连接该主机，并打开Shell会话，请补全下列代码。

```
import .....  
ssh = paramiko.SSHClient()  
ssh.set_missing_host_key_policy(paramiko.....)  
ssh.....(hostname=_____, username=_____, password=_____,  
look_for_keys=_____)  
# 打开交互式 Shell  
cli = ssh.....()
```

目录

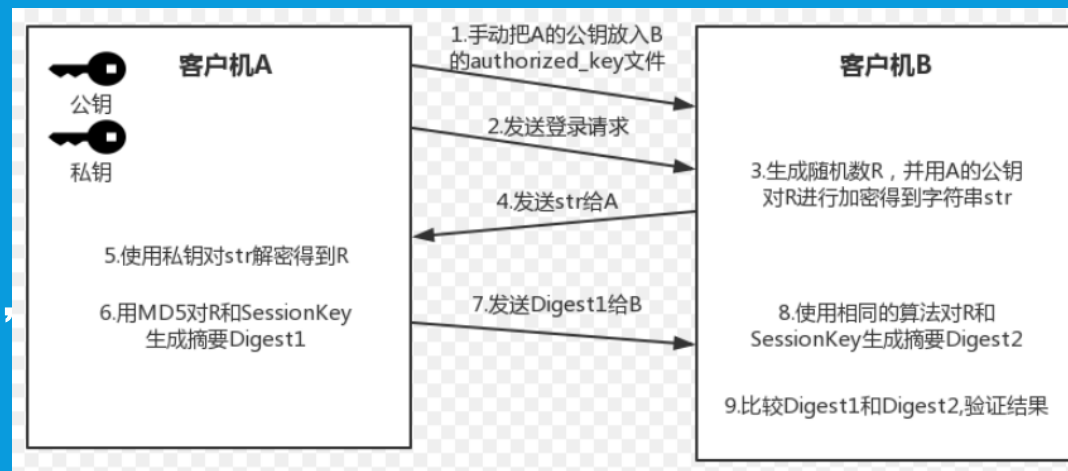
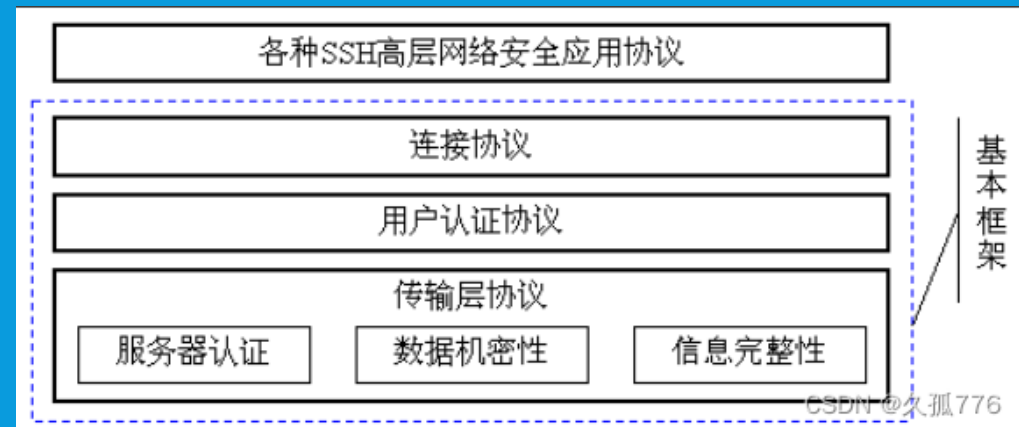
1. 网络设备巡检概念
2. paramiko模块
3. 网络设备配置SSH服务
4. 实训4-使用python脚本实现自动化巡检

3.网络设备配置SSH服务

- 参考教材3.4.1节内容，以拓扑图上路由器SZ1为例，开启SSH服务。

3.1 什么是SSH

1. SSH (Secure Shell) 是一种网络协议，用于在不安全的网络中安全地传输数据。它是一种加密协议，可以保护数据在传输过程中不被窃取、篡改或伪造。SSH协议最初是由芬兰的Tatu Ylonen开发的，现在已经成为了一种标准的网络协议。
2. SSH协议的组成SSH协议由三个部分组成：传输层协议、用户认证协议和连接协议。
3. 传输层协议：传输层协议是SSH协议的核心部分，它负责加密和解密数据，以及保证数据在传输过程中的完整性和机密性。传输层协议使用了一些加密算法，如DES、3DES、AES等，以及一些消息认证码算法，如HMAC、MD5等。
4. 用户认证协议：用户认证协议用于验证用户的身份，以确保只有授权的用户才能访问系统。用户认证协议支持多种认证方式，如口令认证、公钥认证、Kerberos认证等。
5. 连接协议：连接协议用于建立和维护SSH连接。连接协议支持多种连接方式，如TCP连接、UDP连接等。



3.2 华为设备的SSH基本配置命令

[R1]aaa //进入aaa (aaa为认证授权审计)

[R1-aaa]local-user yiqing password cipher huawei123 //在本地创建用户yiqing，密码为huawei123

[R1-aaa]local-user yiqing privilege level 15 //设置用户级别为15

[R1-aaa]local-user yiqing service-type ssh //设置用户登陆服务类型为ssh

[R1]ssh user yiqing authentication-type password //设置此用户登陆为密码认证

[R1]stelnet server enable //开启ssh服务

[R1]user-interface vty 0 4 //进入vty接口

[R1-ui-vty0-4]authentication-mode aaa //认证采用aaa认证

[R1-ui-vty0-4]protocol inbound ssh //vty允许ssh进行登陆

R2:[R2]ssh client first-time enable

//用来使能SSH客户端首次认证

```
[R2]stelnet 192.168.1.1
Please input the username:yiqing
Trying 192.168.1.1 ...
Press CTRL+K to abort
Connected to 192.168.1.1 ...
The server is not authenticated. Continue to access it? (y/n)[n]:y
Jun 21 2023 13:28:51-08:00 R2 %%01SSH/4/CONTINUE_KEYEXCHANGE(1)[0]:The server ha
d not been authenticated in the process of exchanging keys. When deciding whethe
r to continue, the user chose Y.
[R2]
Save the server's public key? (y/n)[n]:y
The server's public key will be saved with the name 192.168.1.1. Please wait...

Jun 21 2023 13:28:54-08:00 R2 %%01SSH/4/SAVE_PUBLICKEY(1)[1]:when deciding wheth
er to save the server's public key 192.168.1.1, the user chose Y.
[R2]
Enter password:
<R1>
<R1>
<R1>
<R1>sys
Enter system view, return user view with Ctrl+Z.
```

3.3 华为设备的SSH配置验证

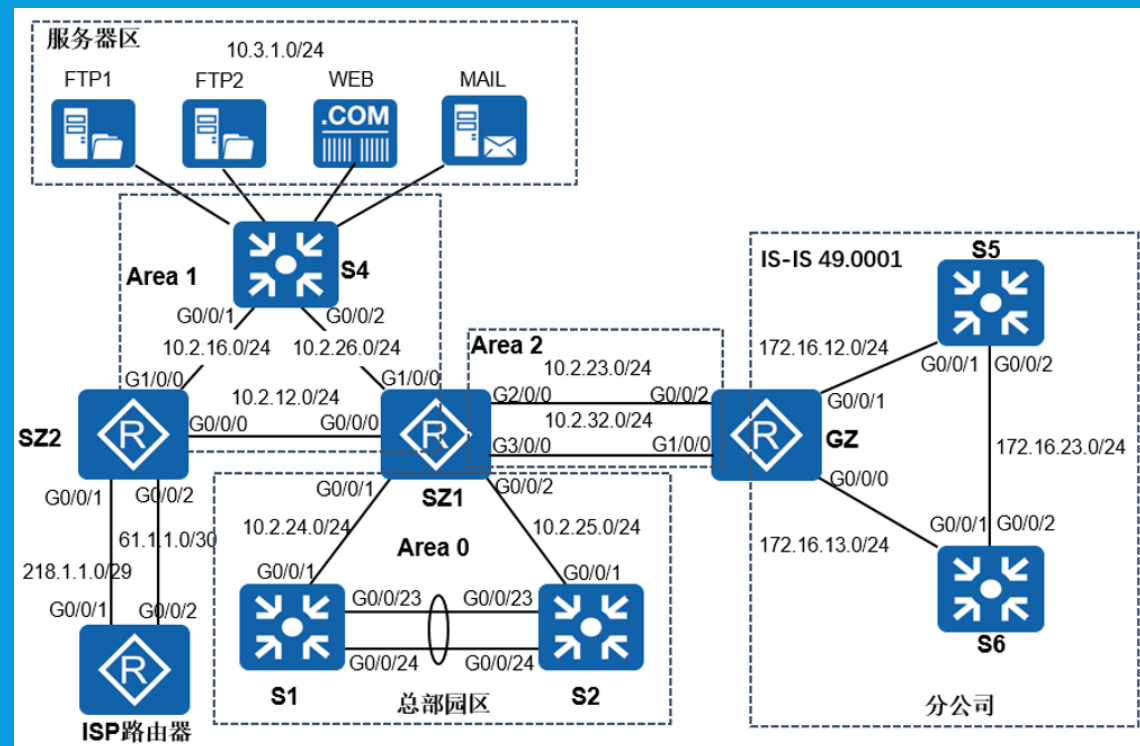
[RZ1] display ssh server status

```
[SZ1]stelnet server enable
Info: Succeeded in starting the STELNET server.
[SZ1]
[SZ1]ssh user python authentication-type password
Authentication type setted, and will be in effect next time
[SZ1]
[SZ1]user-interface vty 0 4
[SZ1-ui-vty0-4]
[SZ1-ui-vty0-4]authentication-mode aaa
[SZ1-ui-vty0-4]
[SZ1-ui-vty0-4]user privilege level 15
[SZ1-ui-vty0-4]
[SZ1-ui-vty0-4]protocol inbound ssh
[SZ1-ui-vty0-4]
[SZ1-ui-vty0-4]q
[SZ1]displ
[SZ1]display ssh server status
SSH version                               :1.99
SSH connection timeout                   :60 seconds
SSH server key generating interval       :0 hours
SSH Authentication retries                :3 times
SFTP Server                             :Disable
Stelnet server                           :Enable
[SZ1]
```

4.实训4-（三）：使用python脚本实现网络设备自动化巡检

【任务目标】参考教材3.4节内容，基于指导教师给的网络拓扑图Ensp文件，完成下列操作：

- （1）配置并验证SSH服务端。
- （2）使用paramiko登录设备。
- （3）自动执行网络巡检的各项命令。



实验效果截图

```
<SZ1>system-view
Enter system view, return user view with Ctrl+Z.
[SZ1]aaa
[SZ1-aaa]local-user yunwei001 password cipher Huawei@123
Info: Add a new user.
[SZ1-aaa]local-user yunwei001 service-type telnet
[SZ1-aaa]quit
[SZ1]user-interface vty 0 4
[SZ1-ui-vty0-4]authentication-mode aaa
[SZ1-ui-vty0-4]user privilege level 3
[SZ1-ui-vty0-4]protocol inbound telnet
[SZ1-ui-vty0-4]quit
[SZ1]ntp enable
    Info:NTP service is already started
[SZ1]ntp-service unicast-server 61.1.1.2
[SZ1]snmp-agent
[SZ1]snmp-agent sys-info version v3
[SZ1]snmp-agent mib-view nt include iso
[SZ1]snmp-agent mib-view rd include iso
[SZ1]snmp-agent mib-view wt include iso
[SZ1]snmp-agent group v3 group01 privacy read-view rd write-view wt notify-view nt
[SZ1]snmp-agent usm-user v3 user01 group01 authentication-mode sha Huawei@123 pr ivacy-mode aes128 Huawei@123
[SZ1]
[SZ1]
设备 10.2.12.1 已经配置完成！
telnet %s 10.2.12.2
```

```
[SZ1]dis current-configuration
[V200R003C00]
#
 sysname SZ1
#
 board add 0/1 1GEC
 board add 0/2 1GEC
 board add 0/3 1GEC
#
 snmp-agent local-engineid 800007DB0300000000000000
 snmp-agent sys-info version v3
 snmp-agent group v3 group01 privacy read-view rd write-view wt notify-view nt
 snmp-agent mib-view nt include iso
 snmp-agent mib-view rd include iso
 snmp-agent mib-view wt include iso
 snmp-agent usm-user v3 user01 group01 authentication-mode sha 4705BD8CE37634ECB4D7DE1D59E200E00DFF5B06
 privacy-mode aes128 4705BD8CE37634ECB4D7DE1D59E200E00DFF5B06
 snmp-agent
#
 clock timezone China-Standard-Time minus 08:00:00
#
 portal local-server load flash:/portalpage.zip
#
 drop illegal-mac alarm
#
 ntp-service unicast-server 61.1.1.2
[SZ1]

Please check whether system data has been changed, and save data in time

Configuration console time out, please press any key to log on
```