一：ubuntu

1. 新版本的prometheus不支持ubuntu 18。请将ubuntu升级到20.04.6 LTS
2. 正常编译即可

二：centos7.9

1、请先检查gcc的版本是否低于5，如果是请先升级，可以使用以下命令进行升级：

sudo yum install centos-release-scl

sudo yum install devtoolset-7-gcc\*

scl enable devtoolset-7 bash

检查gcc,一般情况下都能成功。个别centos会不成功， /opt/rh/devtoolset-7下没有生成enable文件。具体可以咨询南京开发人员。

[root@centos7 include]# gcc -v

.......

gcc version 7.3.1 20180303 (Red Hat 7.3.1-5) (GCC)

2、将scl enable devtoolset-7 bash写入 /etc/profile

执行以下命令，打开/etc/profile文件：

sudo vi /etc/profile

在文件末尾添加如下内容：

scl enable devtoolset-7 bash

或

source /opt/rh/devtoolset-7/enable

保存并退出文件，并执行以下命令，应用修改后的配置：

source /etc/profile

3、检查ssl版本，如果为1.0.2版本，则不支持TLS\_server\_method等函数 （新版本sbi支持ssl加密），需要升级

wget https://www.openssl.org/source/openssl-1.1.1k.tar.gz --no-check-certificate

tar -zxf openssl-1.1.1k.tar.gz

cd openssl-1.1.1k

./config --prefix=/usr/local/openssl

make -j 2

make intall

备份老的

mv /usr/bin/openssl /usr/bin/openssl.old

mv /usr/lib64/libssl.so /usr/lib64/libssl.so.old

创建新的软连接

ln -s /usr/local/openssl/bin/openssl /usr/bin/openssl

ln -s /usr/local/openssl/lib/libssl.so /usr/lib64/libssl.so

ln -s /usr/local/openssl/include/openssl /usr/include/openssl

注：网上一般还有这一步：echo "/usr/local/openssl/lib" >> /etc/ld.so.conf，注意不需要这一步。不然会有libcrypto.so库的问题

检查,如果是带有FIPS的1.1.1版本，则ok

[root@centos7 lib]# openssl version

OpenSSL 1.1.1k FIPS 25 Mar 2021

添加如下命令（可以放到/etc/profile中），确保程序能够找到1.1.1版本的openssl库。

export PKG\_CONFIG\_PATH=/usr/local/openssl/lib/pkgconfig:$PKG\_CONFIG\_PATH

4、更新curl库。默认curl7.52.1不支持1.1.1的openssl库。需要更新到7.68.0及以上版本。

否则SSL协商时会报错：SSL\_accept failed [error:00000000:lib(0):func(0):reason(0)]

检查当前curl版本：

[root@localhost usr]# curl --version

curl 7.52.1 (x86\_64-pc-linux-gnu) libcurl/7.52.1 **OpenSSL/1.0.2k** zlib/1.2.7 nghttp2/1.33.0

更新步骤：

wget [http://curl.haxx.se/download/curl-7.68.0.tar.gz](http://curl.haxx.se/download/curl-7.68.0.tar.gz" \t "_blank)

tar xvf curl-7.68.0.tar.gz

cd curl-7.68.0

#加--disable-werror是因为某些环境编译会将告警认为是error导致编译不成功

./configure --disable-werror

make -j 8

make install

[root@localhost test]# curl --version

curl 7.68.0 (x86\_64-pc-linux-gnu) libcurl/7.68.0 **OpenSSL/1.1.1k** zlib/1.2.7 nghttp2/1.33.0

1. 升级libmicrohttpd库到libmicrohttpd-0.9.55及以上。（否则metrics应答消息）

当前版本为libmicrohttpd-0.9.40

wget <https://ftp.gnu.org/gnu/libmicrohttpd/libmicrohttpd-0.9.55.tar.gz>

./configure

make -j 8

make install

6、为了避免老的编译临时文件的影响，先删除旧的build目录，然后重新构建build目录

$ meson build --prefix=`pwd`/install

#确认下openssl库是否正确

Using 'PKG\_CONFIG\_PATH' from environment with value: '/usr/local/openssl/lib/pkgconfig:/usr/local/lib/pkgconfig/:'

Run-time dependency libssl found: YES 1.1.1k

Using 'PKG\_CONFIG\_PATH' from environment with value: '/usr/local/openssl/lib/pkgconfig:/usr/local/lib/pkgconfig/:'

Run-time dependency libcrypto found: YES 1.1.1k

$ ninja -C build

7、测试SBI的TLS加密时注意：

NRF会出现如下错误：SSL: certificate subject name (nrf.localdomain) does not match target host name '127.0.0.10'

解决1：各NF在配置支持加密的同时，可将no\_verify设置为true，即验证证书时忽略host验证。

no\_tls: false

no\_verify: true

解决2：各NF配置NRF时，采用域名nrf.localdomain而不是IP

nrf:

sbi:

- addr:

# - 127.0.0.10

- nrf.localdomain

参考：[https://blog.csdn.net/u013919153/article/details/119738284](https://blog.csdn.net/u013919153/article/details/119738284" \t "_blank)