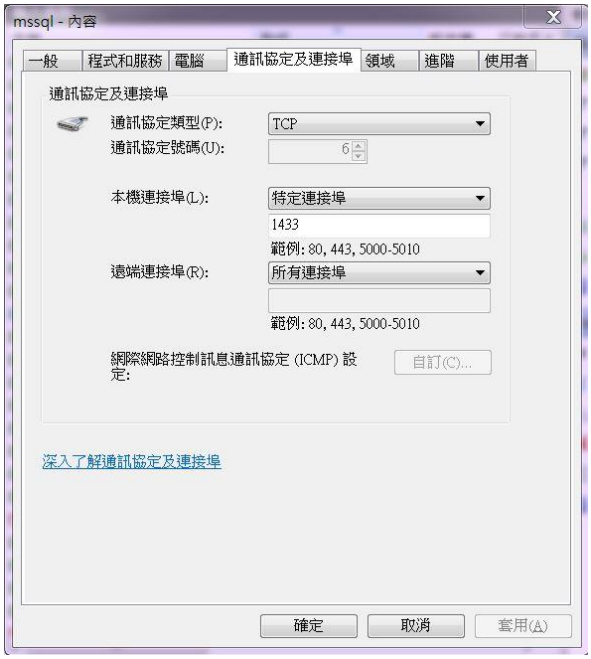
1-0. Windows 防火牆設定

開始 > 控制台 > Windows 防火牆 > 進階設定 > 輸入規則 > 新增規則 > 連接阜(O) > 特定連接阜 1433 > 允許連線(A)> 下一步 > … > 完成

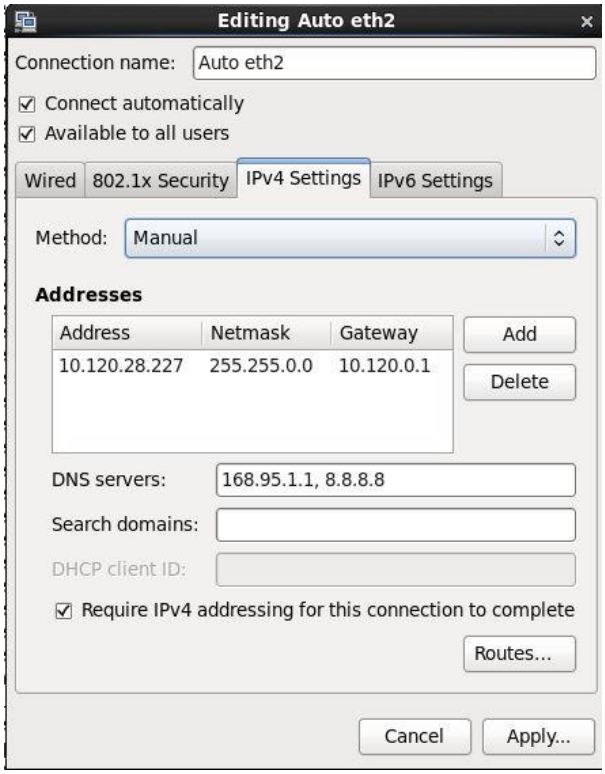


1-1. Download PieTTY

主機名稱或 IP 位置 : 10.120.28.xxx

可從 windows 連進 linux 的 terminal

http://ntu.csie.org/~piaip/pietty/



2-0. Installing the Microsoft SQL Server JDBC Driver

```
### Download the MSSQL Server JDBC driver here and copy it
to /var/lib/sqoop/ directory.

$ tar -xf sqljdbc_4.0.2206.100_cht.tar.gz
$ sudo cp sqljdbc4.jar /var/lib/sqoop
```

3-1. 在 MSSQL 建立一個 view

```
### 包含 1808 部電影的前三個演員和第一週票房和上映年

create view information
AS
SELECT DISTINCT w.pkno, ROUND(LOG (w.weekendgross), 5) AS gross,
ca.priority4c AS sort, ca.cast, w.year
FROM dbo.convHistRateByYear AS b
INNER JOIN
          (SELECT pkno
           FROM dbo.boxoffice
           GROUP BY pkno
           HAVING (SUM(weekendgross) >= 300000)) AS s
ON b.pkno = s.pkno
INNER JOIN
           (SELECT pkno, title, weekendgross, CONVERT(char(4), date) AS year
            FROM dbo.boxoffice
            WHERE (weekend = 1)) AS w
ON s.pkno = w.pkno
INNER JOIN dbo.cast_pri AS ca
ON b.pkno = ca.pkno
WHERE   (0 < ca.priority4c)
AND (4 > ca.priority4c)
```

3-2. Import to HDFS

```
### 將 local 的表格放入 hdfs

$ hadoop fs -put table.txt ./hdfs/table.txt

### 將 sqlserver 表格放入 hdfs
$ sqoop import --connect
"jdbc:sqlserver://10.120.28.27:1433;username=sa;password=passw0rd;database=I
MDB" \
--table information -m 1
```

```
$ hadoop fs -cat ./information/part-m-00000
```

3-2. Import to Hive

```
### 將 sqlserver 表格放入 hive

$ sqoop import --connect
"jdbc:sqlserver://10.120.28.27:1433;username=sa;password=passw0rd;database=I
MDB" \

--table information -m 1 --hive-import
> SELECT * FROM information
```

3-3. Export from Hive

先在 sqlserver 建立好 information_hive 表格

資料行名稱	資料類型	允許 Null
pkno	nvarchar(MAX)	☑
gross	float	☑
sort	int	☑
cast	nvarchar(MAX)	☑
year	nvarchar(MAX)	☑
		☐


```
$ sqoop export --connect
"jdbc:sqlserver://10.120.28.27:1433;username=sa;password=passw0rd;database=I
MDB" -m -1 --table information_hive --export-dir
/user/hive/warehouse/information --input-fields-terminated-by '\0001'
```

```
15/01/09 04:55:34 INFO mapreduce.Job:  map 0% reduce 0%
15/01/09 04:55:41 INFO mapreduce.Job:  map 25% reduce 0%
15/01/09 04:55:50 INFO mapreduce.Job:  map 75% reduce 0%
15/01/09 04:55:59 INFO mapreduce.Job:  map 100% reduce 0%
15/01/09 04:55:59 INFO mapreduce.Job: Job job_1420769802002_0022 completed successfully
15/01/09 04:55:59 INFO mapreduce.Job: Counters: 30
        File System Counters
                FILE: Number of bytes read=0
                FILE: Number of bytes written=515116
                FILE: Number of read operations=0
                FILE: Number of large read operations=0
                FILE: Number of write operations=0
                HDFS: Number of bytes read=445617
                HDFS: Number of bytes written=0
                HDFS: Number of read operations=22
                HDFS: Number of large read operations=0
                HDFS: Number of write operations=0
        Job Counters
                Launched map tasks=4
                Data-local map tasks=4
                Total time spent by all maps in occupied slots (ms)=45709
                Total time spent by all reduces in occupied slots (ms)=0
                Total time spent by all map tasks (ms)=45709
                Total vcore-seconds taken by all map tasks=45709
                Total megabyte-seconds taken by all map tasks=46806016
        Map-Reduce Framework
                Map input records=10844
                Map output records=10844
                Input split bytes=835
                Spilled Records=0
                Failed Shuffles=0
                Merged Map outputs=0
                GC time elapsed (ms)=200
                CPU time spent (ms)=5870
                Physical memory (bytes) snapshot=653389824
                Virtual memory (bytes) snapshot=3571458048
                Total committed heap usage (bytes)=622854144
        File Input Format Counters
                Bytes Read=0
        File Output Format Counters
                Bytes Written=0
15/01/09 04:55:59 INFO mapreduce.ExportJobBase: Transferred 435.1729 KB in 32.7134 seconds (13.3026 KB/sec)
15/01/09 04:55:59 INFO mapreduce.ExportJobBase: Exported 10844 records.
[cloudera@quickstart hadoop_streaming_101]$ echo $SQOOP_HOME

[cloudera@quickstart hadoop_streaming_101]$ whereis sqoop
sqoop: /usr/bin/sqoop /etc/sqoop /usr/lib/sqoop /usr/share/man/man1/sqoop.1.gz
[cloudera@quickstart hadoop_streaming_101]$
```

4-0. 做一個所有演員在 2004~2014 年的電影票房的表(用"-"隔開)

```
$ hadoop jar /usr/lib/hadoop-mapreduce/hadoop-streaming.jar \
-files ./map.py,./reduce.py \
-input information/part* \
-output cast_step1 \
-mapper map.py \
-reducer reduce.py
```

4-1. 將演員當年度票房為 0 的地方用前一年的票房補上

```
$ hadoop jar /usr/lib/hadoop-mapreduce/hadoop-streaming.jar \
-files ./mapsort.py \
-input cast_step1/part* \
-output cast_step2 \
-mapper mapsort.py \
-numReduceTasks 0
```

4-2. 將演員票房表丟到本機端並建成一個 table.txt 的檔案

```
$ hadoop fs -copyToLocal cast_step2/ cast_table
$ cat cast_table/part* > table.txt
```

4-3. 將 1808 部電影的三個演員權重算出來(用"-"隔開)

```
$ hadoop jar /usr/lib/hadoop-mapreduce/hadoop-streaming.jar \
-files ./weight_map.py,./weight_reduce.py,./table.txt \
-input information/part* \
-output cast_step3 \
-mapper weight_map.py \
-reducer weight_reduce.py
```

Others.

```
### 如果 MR 執行失敗請先刪掉當時 output 的路徑
$ hadoop fs -rm -r cast_step3(路徑名)

### 看結果
$ hadoop fs -cat cast_step3/part-00000
$ hadoop fs -cat cast_step3/part-00001
```