### 顺序

先装WSFC并配置；

再装sqlserver2019【企业版密钥:HMWJ3-KY3J2-NMVD7-KG4JR-X2G8G】

### 无域版AG搭建

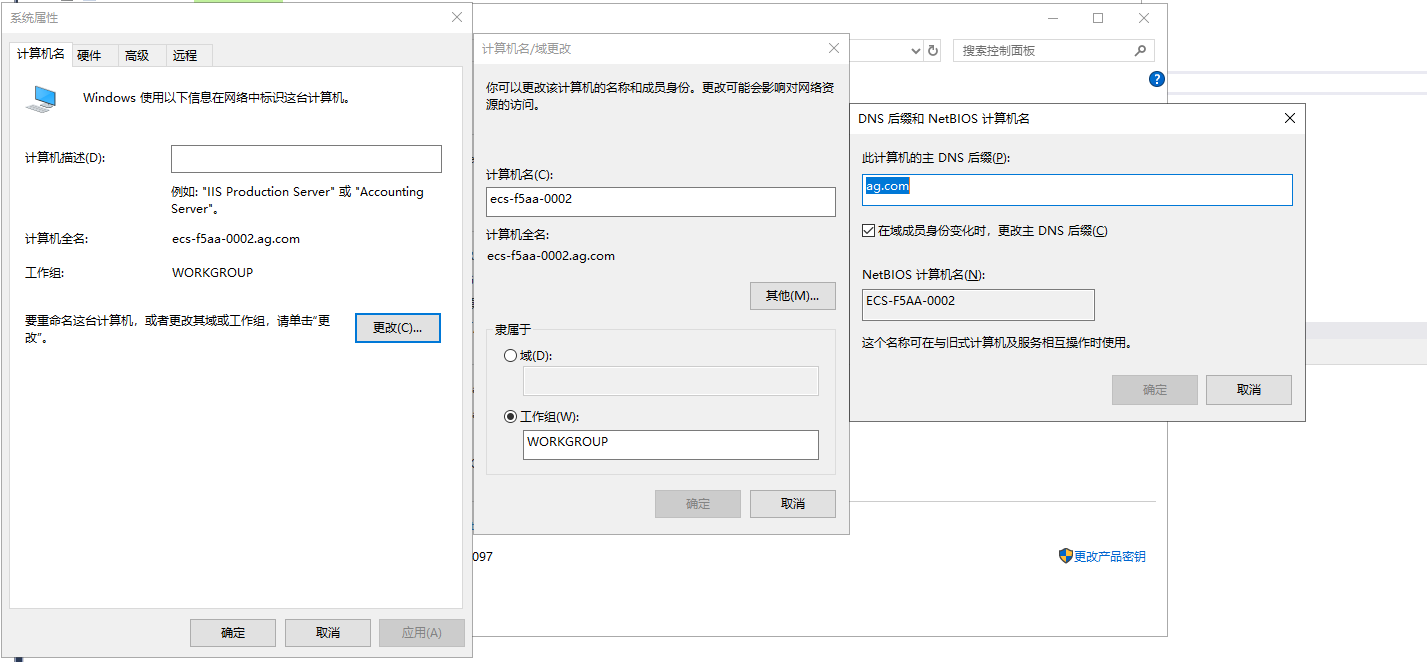
<https://blog.csdn.net/activeRabbit/article/details/107906083>

#### 所有服务器:安装故障转移群集功能



#### 所有服务器：计算机名称修改，加DNS后缀

实践中只加了DNS后缀，通过点击其他添加；可以将计算机名称先修改成便于记住的名称。



#### 所有服务器：修改hosts文件

|  |
| --- |
| 192.168.0.66 ecs-f5aa-0002.ag.com  192.168.0.173 ecs-f5aa-0001.ag.com  192.168.0.157 WIN-TQ09P8PS8ED.ag.com  192.168.0.201 ag\_cluster 虚拟的集群IP，访问时用这个地址 |

#### 云服务器配置虚拟IP

【真机不需要配置】

由于云服务器是虚拟机虚拟出来的，虚拟IP未配置则不会随这节点的转移而转移，即不会生效，故需要配置一下。以下是华为云虚拟IP配置的说明。

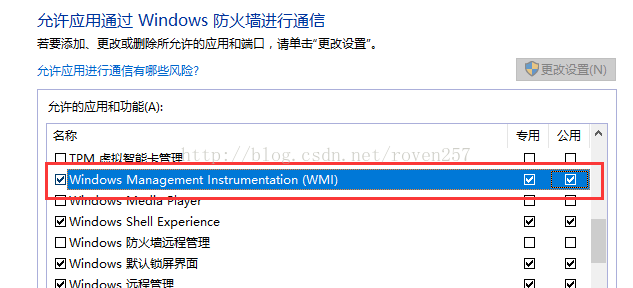
|  |
| --- |
| <https://console.huaweicloud.com/vpc/?region=cn-north-4#/vpcs>  虚拟私有云 |

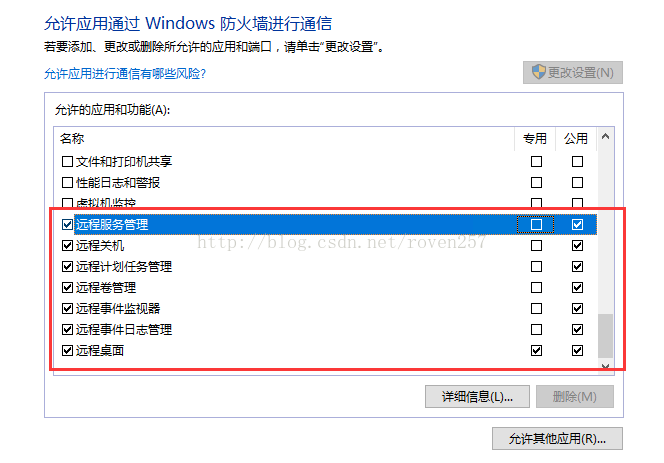
#### 所有服务器：关闭防火墙

设置允许应用或功能通过windows防火墙，这里我为了图省事直接把防火墙关了。

正式环境：

设置允许应用或功能通过防火墙，按照下面图中红框框出的地方设置，注意选项后面打勾的位置。





#### 所有服务器：修改DNS

首选DNS服务器修改为 集群IP



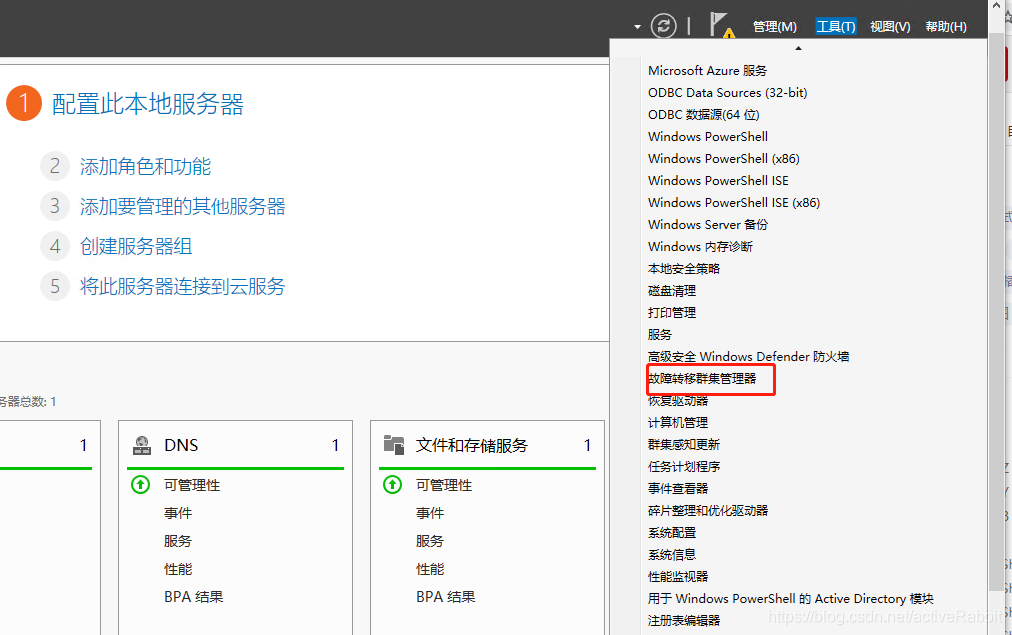
#### 所有服务器：新增管理员账户，名称密码需一致，且隶属于管理员组

为安全起见还是新增专用于AG的账户来操作；如果用建立管理员组的普通用户来创建集群，两节点用户名密码必须相同；还需要在两节点注册表添加 LocalAccountTokenFilterPolicy为1【此设置会留下隐患】[故障转移集群添加时提示无管理员权限解决方案](https://blog.csdn.net/ken119/article/details/103944447) [高级域渗透技术之传递哈希已死-LocalAccountTokenFilterPolicy万岁](https://blog.csdn.net/systemino/article/details/89716729)。

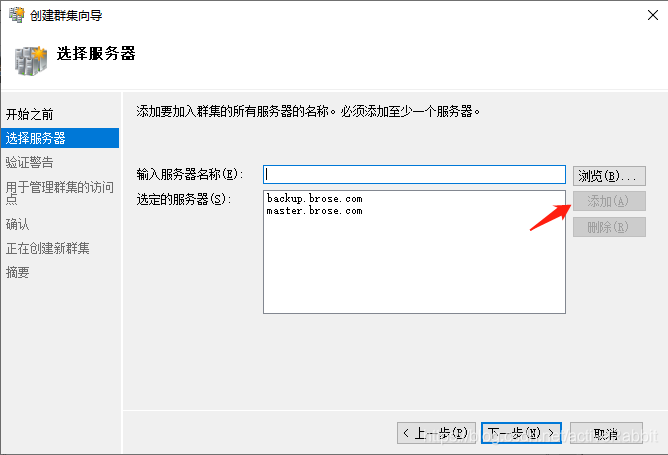




#### 任意一台服务器：新建故障转移集群



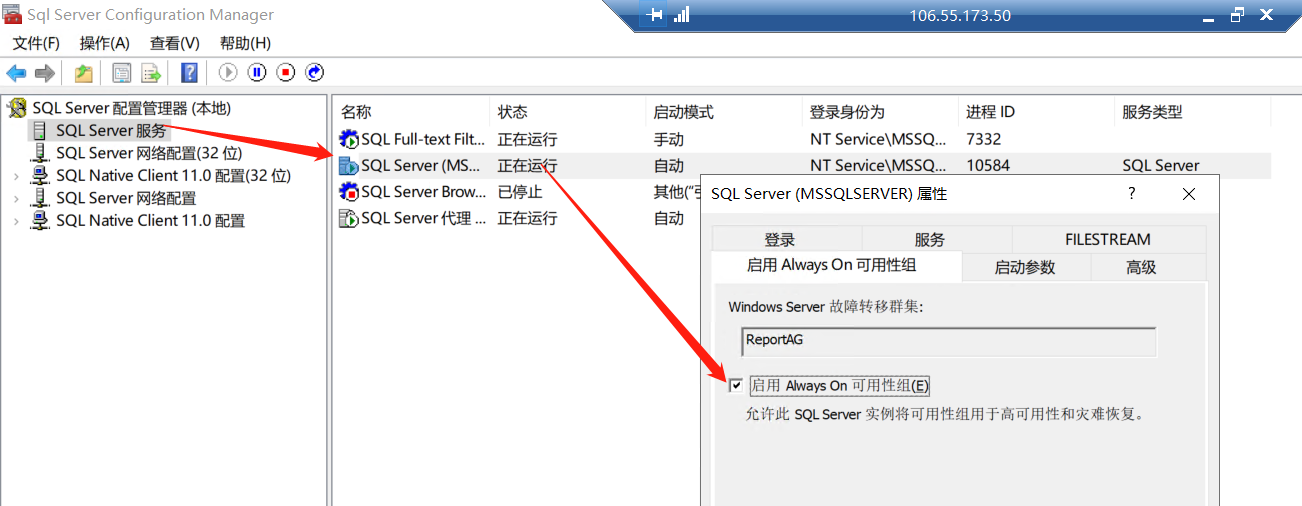
**分别输入master、backup添加，效果如图**





#### 所有服务器：启用Always On可用性组

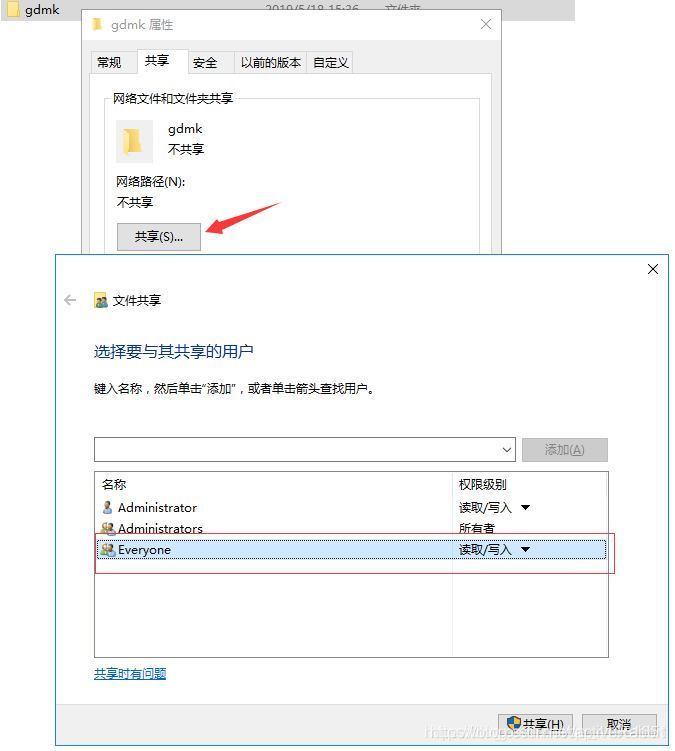
启用后需重启一下SQL Server服务。



#### 所有服务器：设置共享文件夹

在C盘中新建一个gdmk文件夹(两台都要建)

设置共享人为Everyone,修改权限级别为读取/写入



#### 所有服务器：确定sqlserver的server name是否为计算机名称

1. 加入AG组的数据库需要完整备份。
2. 端点Url填的也是计算机名称



1. 由于sqlserver是以服务器hostname为服务器实例名称，所以需要修改每个服务器计算机名称和实例名称一致，且需保证名称不重复；若是在安装了SQL Server后才修改的hostname，则需要在执行下列sql脚本，再重启服务器上的sqlserver服务

|  |
| --- |
| select @@servername  Sp\_dropserver 'localhost'--为第一条sql查出的服务器就名称  Sp\_addserver 'server150','local' --server150为新的服务器名称，local不需修改 |

#### 所有服务器：在数据库里面安装证书

##### 主节点上执行

|  |
| --- |
| :CONNECT 192.168.0.66 -U sa -P zjzt#123456  GO  USE master;  GO  --创建主密钥  CREATE MASTER KEY ENCRYPTION BY PASSWORD = 'zjzt@123456';  --创建证书 这里有两个"CERT1",建议每台机器都改成不同的名称以作区分  CREATE CERTIFICATE CERT1 WITH SUBJECT = 'CERT1'  --把刚才创建的证书备份到文件 这里也有两个CERT1要改  BACKUP CERTIFICATE CERT1 TO FILE = 'C:\gdmk\CERT1.cer';  --创建终结点,设为证书验证 这里第二行有1个CERT1要改  CREATE ENDPOINT [ag\_endpoint] AUTHORIZATION [sa] STATE=STARTED AS TCP (LISTENER\_PORT = 5022, LISTENER\_IP = ALL)  FOR DATA\_MIRRORING (ROLE = ALL,AUTHENTICATION = CERTIFICATE CERT1, ENCRYPTION = REQUIRED ALGORITHM AES)  GO  :CONNECT 192.168.0.173 -U sa -P zjzt#123456  GO  USE master;  GO  --创建主密钥  CREATE MASTER KEY ENCRYPTION BY PASSWORD = 'zjzt@123456';  --创建证书 这里有两个"CERT2",建议每台机器都改成不同的名称以作区分  CREATE CERTIFICATE CERT2 WITH SUBJECT = 'CERT2'  --把刚才创建的证书备份到文件 这里也有两个CERT2要改  BACKUP CERTIFICATE CERT2 TO FILE = 'C:\gdmk\CERT2.cer';  --创建终结点,设为证书验证 这里第二行有1个CERT2要改  CREATE ENDPOINT [ag\_endpoint] AUTHORIZATION [sa] STATE=STARTED AS TCP (LISTENER\_PORT = 5022, LISTENER\_IP = ALL)  FOR DATA\_MIRRORING (ROLE = ALL,AUTHENTICATION = CERTIFICATE CERT2, ENCRYPTION = REQUIRED ALGORITHM AES)  GO  :CONNECT 192.168.0.157 -U sa -P zjzt#123456  GO  USE master;  GO  --创建主密钥  CREATE MASTER KEY ENCRYPTION BY PASSWORD = 'zjzt@123456';  --创建证书 这里有两个"CERT3",建议每台机器都改成不同的名称以作区分  CREATE CERTIFICATE CERT3 WITH SUBJECT = 'CERT3'  --把刚才创建的证书备份到文件 这里也有两个CERT3要改  BACKUP CERTIFICATE CERT3 TO FILE = 'C:\gdmk\CERT3.cer';  --创建终结点,设为证书验证 这里第二行有1个CERT3要改  CREATE ENDPOINT [ag\_endpoint] AUTHORIZATION [sa] STATE=STARTED AS TCP (LISTENER\_PORT = 5022, LISTENER\_IP = ALL)  FOR DATA\_MIRRORING (ROLE = ALL,AUTHENTICATION = CERTIFICATE CERT3, ENCRYPTION = REQUIRED ALGORITHM AES)  GO |

##### 将每个节点的证书copy至其它两台机子

##### 各节点相互载入对方证书

主节点

|  |
| --- |
| USE master;  GO  CREATE CERTIFICATE CERT2 FROM FILE = 'C:\gdmk\CERT2.cer';  CREATE CERTIFICATE CERT3 FROM FILE = 'C:\gdmk\CERT3.cer'; |

第二台机子

|  |
| --- |
| USE master;  GO  CREATE CERTIFICATE CERT1 FROM FILE = 'C:\gdmk\CERT1.cer';  CREATE CERTIFICATE CERT3 FROM FILE = 'C:\gdmk\CERT3.cer'; |

第三台机子

|  |
| --- |
| GO  USE master;  GO  CREATE CERTIFICATE CERT1 FROM FILE = 'C:\gdmk\CERT1.cer';  CREATE CERTIFICATE CERT2 FROM FILE = 'C:\gdmk\CERT2.cer'; |

#### 利用HAProxy代理SQL Server【Extra】

【前提：所有服务器上的同名库能同步且能读写】

所有服务器：保证SQL Server代理服务已开启。

准备一台Linux机器

参考[利用HAProxy代理SQL Server](https://blog.csdn.net/yujin2010good/article/details/76034743)

yum安装包

|  |
| --- |
| yum install haproxy -y |

清空配置文件

|  |
| --- |
| true > /etc/haproxy/haproxy.cfg |

编辑配置文件

|  |
| --- |
| vim /etc/haproxy/haproxy.cfg |

这里配置了两个sqlserver的集群，一个windows的，一个容器的

|  |
| --- |
| defaults  mode http  option dontlognull  #option http-server-close  #option forwardfor except 127.0.0.0/8  option redispatch  retries 3  timeout http-request 10s  timeout connect 10s  timeout client 1m  timeout server 1m  timeout http-keep-alive 10s  timeout check 10s  maxconn 6000  listen mssql  bind 0.0.0.0:1433  mode tcp  option tcplog  balance roundrobin  server mssqldb1 192.168.0.157:1433 weight 1 maxconn 6000 check port 1433 inter 2000 rise 2 fall 2  server mssqldb2 192.168.0.173:1433 weight 1 maxconn 6000 check port 1433 inter 2000 rise 2 fall 2  listen mssql\_linux  bind 0.0.0.0:1500  mode tcp  option tcplog  balance roundrobin  server db1 192.168.0.66:1501 weight 1 maxconn 6000 check port 1501 inter 2000 rise 2 fall 2  server db2 192.168.0.66:1502 weight 1 maxconn 6000 check port 1502 inter 2000 rise 2 fall 2  server db2 192.168.0.66:1503 weight 1 maxconn 6000 check port 1503 inter 2000 rise 2 fall 2  listen monitor  bind 0.0.0.0:8100  mode http  option httplog  stats enable  stats uri /stats  stats auth admin:123456  stats refresh 5s |

检测配置文件的正确性

|  |
| --- |
| haproxy -f /etc/haproxy/haproxy.cfg -c |

Haproxy服务停止后启动

|  |
| --- |
| systemctl stop haproxy  systemctl start haproxy  #另一个启动命令  /usr/sbin/haproxy -f /etc/haproxy/haproxy.cfg |

检查haproxy是否在监听

|  |
| --- |
| netstat -lntp |

查看群集状态

|  |
| --- |
| http://114.115.130.239:8100/stats |