

CVB Installation Doc for R2

Installation Architecture

Due to the hardware source limitation, Connected Vehicle Blueprint is deployed in three Virtual Machines in Amazon Web Service.

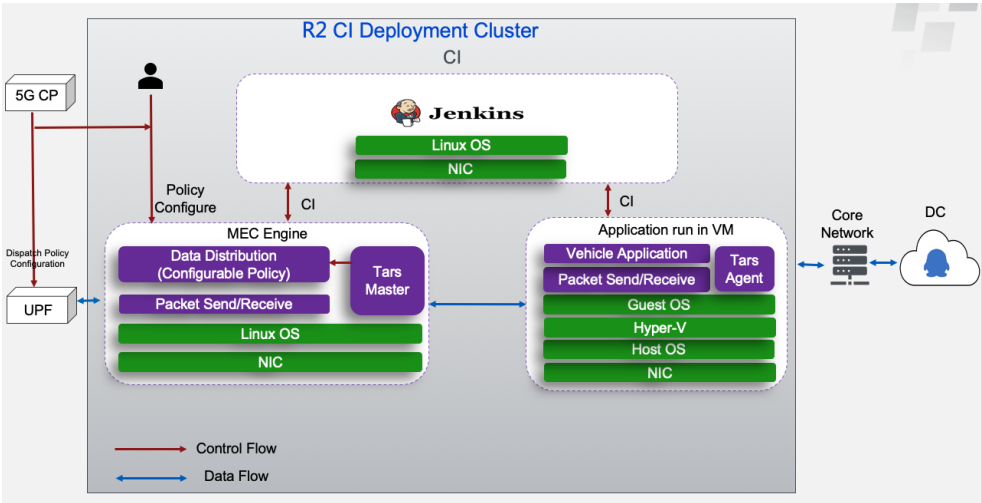
(Visit [CI Lab Environment Setup](#) if you want to set up connected vehicle blueprint in CI Lab.)

The following picture depicts the deployment architecture.

Server A : Deploy Jenkins.

Server B : Deploy Tars Master Node.

Server C Deploy Tars Slave(Node) and the connected vehicle applications.



Installation Hardware

Server Name	CPU+Memory	Drive	Deployment
Jenkins	A1 8Core * 16G	15G	Jenkins Master
TarsFramework	A1 8Core * 16G	10+50G	TarsFramework
TarsNode	A1 8Core * 16G	10G +20G	TarsNode + Application

Installation Software

- CentOS 7 centos-7_aarch64 - ami-012355fc520b79a12
- mysql Ver 14.14 Distrib 5.6.26, for Linux (aarch64) using EditLine wrapper
- OpenStack: Rocky
- k8s:1.15.0

Installation Step by Step

Step1: Launch Instance in the Cloud

For Tencent Cloud, refer to the following link to apply new instance:

<https://intl.cloud.tencent.com/document/product/213/9384?lang=en>

For AWS A1, apply new instance, refer to the following link to apply new instance:

<https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/concepts.html>

AMI Details

centos-7_aarch64 - ami-012355fc520b79a12

CentOS 7 (aarch64) HVM

Root Device Type: ebs Virtualization type: hvm

Edit AMI

Instance Type

Instance Type	ECUs	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance
a1.2xlarge	-	8	16	EBS only	Yes	Up to 10 Gigabit

Edit instance type

EC2 Dashboard

Events

Tags

Reports

Limits

INSTANCES

Instances

Launch Templates

Spot Requests

Launch Instance

Connect

Actions

Filter by tags and attributes or search by keyword

1 to 3 of 3

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS (IPv4)	IPv4 Public IP	IPv6 IPs	Key Name	Monitoring	Launch T
VM2-TarsNode	i-0c4eb542c86840bf1	a1.2xlarge	us-west-2c	running	2/2 checks ...	None	ec2-34-219-114-7.us-w...	34.219.114.7	-	AkrainoR2	disabled	November
VM1	i-055edb8b744fc9276	a1.2xlarge	us-west-2c	running	2/2 checks ...	None	ec2-34-217-53-6.us-we...	34.217.53.6	-	AkrainoR2	disabled	November
Jenkins	i-01d8465d6e7f96878	a1.2xlarge	us-west-2c	running	2/2 checks ...	None	ec2-54-218-53-101.us-...	54.218.53.101	-	AkrainoR2	disabled	November

Step2: Install Jenkins

Refer to the following link for installing Jenkins Mater and connect to Jenkins Slave.

<https://github.com/qiuxin/Connected-Vehicle>

Jenkins

search

ConnectedVehicle

log out

Jenkins

Nodes

Back to Dashboard

Manage Jenkins

New Node

Configure

Build Queue

No builds in the queue.

Build Executor Status

master

1 idle

2 idle

vm1-TarsFramework

1 idle

vm2-TarsNode

1 idle

S	Name	Architecture	Clock Difference	Free Disk Space	Free Swap Space	Free Temp Space	Response Time
	master	Linux (aarch64)	In sync	4.56 GB	819.94 MB	4.56 GB	0ms
	vm1-TarsFramework	Linux (aarch64)	In sync	45.24 GB	819.94 MB	1.88 GB	3ms
	vm2-TarsNode	Linux (aarch64)	In sync	3.59 GB	819.94 MB	3.59 GB	1ms
Data obtained		39 min	39 min	39 min	39 min	39 min	39 min

Refresh status

Jenkins

SEARCH

ConnectedVehicle

log out

New Item

People

Build History

Project Relationship

Check File Fingerprint

Manage Jenkins

My Views

Credentials

Lockable Resources

New View

Build Queue

No builds in the queue.

Build Executor Status

master

1 Idle

2 Idle

vm1-TarsFramework

1 Idle

vm2-TarsNode

1 Idle

All

S	W	Name	Last Success	Last Failure	Last Duration
		CompileMysql-TarsNode	1 day 4 hr - #1	N/A	32 min
		CompileTarsCode	1 day 17 hr - #2	1 day 17 hr - #1	21 min
		CompileTarsCode-TarsNode	1 day 3 hr - #1	N/A	21 min
		Config.mysql	1 day 18 hr - #1	N/A	14 sec
		Install.Mysql	1 day 19 hr - #1	N/A	32 min
		InstallBas-TarsNode	1 day 4 hr - #1	N/A	2 min 7 sec
		InstallBasicLib	1 day 19 hr - #1	N/A	2 min 3 sec
		InstallNVM	1 day 18 hr - #1	N/A	22 sec
		InstallTarsFramework_TarsNode	1 day 2 hr - #1	N/A	1.3 sec
		InstallTarsFramework	1 day 17 hr - #1	N/A	1.4 sec
		MountDisk	1 day 19 hr - #1	N/A	2.7 sec
		StartTars-TarsNode	N/A	N/A	N/A
		StartTarsFramework	N/A	N/A	N/A
		TestCompileCode	15 hr - #6	19 hr - #3	21 min
		TestConnectVehicleService	1 hr 56 min - #14	N/A	56 ms

Icon: S M L

Legend

RSS for all

RSS for failures

RSS for just latest builds

Step3: Install Tarsframework

Refer to the following link for installing Tarsframework.

<https://github.com/qiuxin/Tars/blob/arm/Install.md>

The following is the picture for Tars Framework website.

TARS

服务管理

运维管理

中文

admin

tars

tarspatch

tarsconfig

tarsnotify

tarslog

tarsstat

tarsproperty

tarsqueryproperty

tarsquerystat

CVApp

HelloService

服务管理

应用配置

服务列表

服务	节点	启用Set	设置状态	当前状态	进程ID	版本	发布时间	操作
tarsconfig	172.31.10.49	不启用	Active	Active	26572		Invalid date	编辑 重启 停止 管理Servant 更多命令
tarslog	172.31.10.49	不启用	Active	Active	26725	52	2019-11-02 12:20:44	编辑 重启 停止 管理Servant 更多命令
tarsnotify	172.31.10.49	不启用	Active	Active	26726	53	2019-11-02 12:21:47	编辑 重启 停止 管理Servant 更多命令
tarspatch	172.31.10.49	不启用	Active	Active	26641		Invalid date	编辑 重启 停止 管理Servant 更多命令
tarsproperty	172.31.10.49	不启用	Active	Active	26744	54	2019-11-02 12:22:39	编辑 重启 停止 管理Servant 更多命令
tarsqueryproperty	172.31.10.49	不启用	Active	Active	26760	55	2019-11-02 12:23:26	编辑 重启 停止 管理Servant 更多命令
tarsquerystat	172.31.10.49	不启用	Active	Active	26768	56	2019-11-02 12:24:36	编辑 重启 停止 管理Servant 更多命令
tarsstat	172.31.10.49	不启用	Active	Active	26782	58	2019-11-02 12:26:07	编辑 重启 停止 管理Servant 更多命令

<

1

>

TARS

服务管理

运维管理

中文

admin

tars

tarspatch

tarsconfig

tarsnotify

tarslog

tarsstat

tarsproperty

tarsqueryproperty

tarsquerystat

CVApp

HelloService

服务管理

发布管理

服务配置

服务监控

特性监控

接口调试

服务列表

服务	节点	应用Set	设置状态	当前状态	进程ID	版本	发布时间	操作
HelloService	172.31.14.160	不启用	Active	Active	7539	61	2019-11-03 07:22:39	编辑 重启 停止 管理Servant 更多命令

服务实时状态

时间	服务ID	线程ID	结果
2019-11-03 07:22:39	CVApp.HelloService_172.31.14.160		patch CVApp.HelloService succ, version 61
2019-11-03 07:22:39	CVApp.HelloService_172.31.14.160	281472855198192	stop
2019-11-03 07:22:39	CVApp.HelloService_172.31.14.160	281472827542000	restart
2019-11-03 06:18:15	CVApp.HelloService_172.31.14.160	281473158367728	stop
2019-11-03 06:18:15	CVApp.HelloService_172.31.14.160	281472855198192	restart
2019-11-03 06:18:14	CVApp.HelloService_172.31.14.160		patch CVApp.HelloService succ, version 60
2019-11-03 04:09:09	CVApp.HelloService_172.31.14.160	281473158367728	restart
2019-11-03 04:09:08	CVApp.HelloService_172.31.14.160		patch CVApp.HelloService succ, version 59

<

1

>

Step4: Tars Node

Refer to the following link for installing Tarsnode and connect Tarsnode to Tarsframework.

<https://github.com/qiuxin/Tars/blob/arm/Install.md>

Step5: Launch CI jobs

Create CI jobs and launch CI jobs in the following way.

The detail of the Jenkins script is depicted in the Test Document.

[CVB Test Doc for R2](#)

Jenkins

ConnectedVehicle | log out

New Item

People

Build History

Project Relationship

Check File Fingerprint

Manage Jenkins

My Views

Credentials

Lockable Resources

New View

Build Queue

No builds in the queue.

Build Executor Status

master

1 idle

2 idle

vm1-TarsFramework

1 idle

vm2-TarsNode

1 idle

All

S	W	Name	Last Success	Last Failure	Last Duration
		ComoleMySQL-TarsNode	1 day 4 hr - #1	N/A	32 min
		ComoleTarsCode	1 day 17 hr - #2	1 day 18 hr - #1	21 min
		ComoleTarsCode-TarsNode	1 day 3 hr - #1	N/A	21 min
		Config_mysql	1 day 18 hr - #1	N/A	14 sec
		Install_Mysql	1 day 19 hr - #1	N/A	32 min
		InstallBao-TarsNode	1 day 4 hr - #1	N/A	2 min 7 sec
		InstallBasicLib	1 day 19 hr - #1	N/A	2 min 3 sec
		InstallKVM	1 day 18 hr - #1	N/A	22 sec
		InstallTarsFramework-TarsNode	1 day 2 hr - #1	N/A	1.3 sec
		InstallTarsFramework	1 day 17 hr - #1	N/A	1.4 sec
		MountDisk	1 day 19 hr - #1	N/A	2.7 sec
		StartTars-TarsNode	N/A	N/A	N/A
		StartTarsFramework	N/A	N/A	N/A
		TestCompileCode	16 hr - #6	19 hr - #3	21 min
		TestConnectVehicleService	1 min 57 sec - #15	N/A	59 ms

Icon: S M L

Legend RSS for all RSS for failures RSS for just latest builds

General | Source Code Management | **Build Triggers** | Build Environment | Build | Post-build Actions

Trigger whenever repository changes occur manually or automatically

- ☐ Build after other projects are built
- ☒ Build periodically

Schedule `* * * * *`

Spread load evenly by using "H 12 *" rather than "* * * * *"**
 Would last have run at Sunday, November 3, 2019 12:00:08 PM UTC; would next run at Monday, November 4, 2019 12:00:08 PM UTC.

- ☐ GitHub hook trigger for GITScm polling
- ☐ Poll SCM

Build Environment

- ☐ Delete workspace before build starts
- ☐ Use secret text(s) or file(s)
- ☐ Abort the build if it's stuck
- ☐ Add timestamps to the Console Output
- ☐ Inspect build log for published Gradle build scans
- ☐ With Art

Build

☒ Execute shell

```
Command
cd /usr/local/robot/testCompileCode
git clone -b km https://github.com/gjuvin/Tars.git
cd /usr/local/robot/testCompileCode/Tars
git submodule update --init --recursive
cd /usr/local/robot/testCompileCode/Tars/framework/build
chmod +w build.sh
./build.sh all
rm -rf /usr/local/robot/testCompileCode/Tars
```

View this list of available environment variables

Save Apply Advanced...