

TripCom Scalability Tests

The EC2 Side Of Things



Daniel Martin

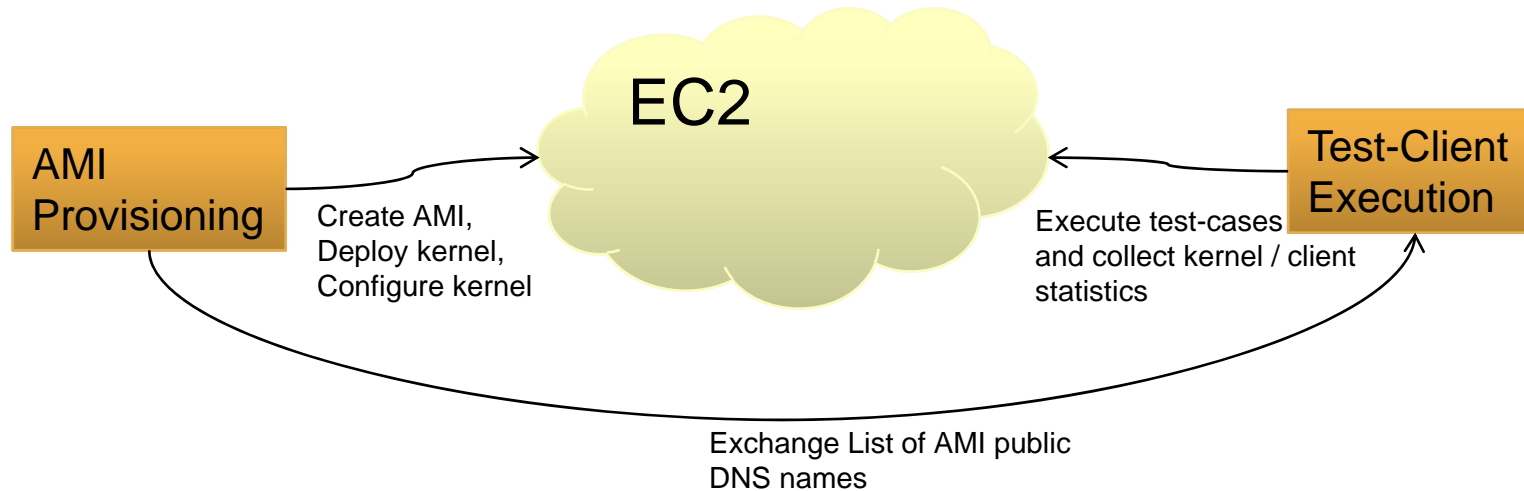
[daniel.martin@iaas.uni-stuttgart.de]

IAAS, University of Stuttgart

September 15th



- High-level Architecture
- Custom-Build Tools
- Sample Deployment Script
- Live Demo
- Organization
- Schedule



- AMI public DNS names are exchanged out-of-band (list of “instance descriptors – see later”) e.g. per email

- Each test-case is a separate directory. The scripts are executed in this directory to persist state of the current ec2 deployment
- `ec2createinstances $1`
 - `$1` is the number of instances to allocate
 - script waits until all requested machines are booted and ready to be accessed
 - creates a file (called “instance descriptor”) for each started instance:
 - filename: Amazon instance ID (e.g. i-ag453jge3)
 - file contents: public DNS name of started machine
- `ec2execremote $1 [-bg]`
 - `$1` is the command to be executed on each ec2 instance described by the “instance descriptor” files in the current directory
 - `-bg` runs the task in background
 - command is run as user “tripcom”

- `ec2execlocal $1 [-bg]`
 - Executes command `$1` once for each running instance, but locally. Variables `$CURRENT_AMI_DNS` and `$CURRENT_AMI_ID` get expanded accordingly
 - Useful e.g. for `scp` commands (see example)
 - command is run as user “tripcom”
- `ec2killall`
 - Kills all ec2 instances described by “instance descriptor” files in the current directory
- Everything else (e.g firewall config, availability zones, login keys, etc) is created and handled automatically by the tools.
- A test-case is a shell-script that uses these tools, e.g.

```
1  #!/usr/bin/env bash
2
3  mkdir testcase1
4  cd testcase1
5  ec2createinstances 2
6  ec2execall "scp ../kernel.zip tripcom@$CURRENT_AMI_DNS:/home/tripcom/kernel.zip"
7  ec2execall "unzip kernel.zip"
```

- Live Demo – if desired

- TripCom Kernel support (FUB / TUW)
 - Provide stable and tested version of the kernel
 - Create .zip file with all dependencies and startup / Management scripts for simple deployment
 - Provide kernel validation script (is the kernel bootstrapped and running correctly? Is the PGrid routing table correct?)
- EC2 Deployment (USTUTT)
 - Start machines on EC2
 - provision machines (deploy kernels, etc.)

- M30: First Version of EC2 deployment tools ready
- M34: First scalability tests, check infrastructure, tooling, test-clients, ...
- M35: Final Version scalability tests
- M36: Report write-up

End of Document