heinlein

Robert Sander <r.sander@heinlein-support.de>

CloudStack & Terraform

Robert Sander <r.sander@heinlein-support.de>

Terraform?

- Provisioning
- Management

- on CloudStack

Robert Sander <r.sander@heinlein-support.de>

- **Heinlein Support**
  - IT Consulting with 24/7 Linux support team (35 staff)
  - Running ISP operations since 1992
  - Vast experience of supporting core IT services for business of all sizes

- 24/7 Emergency Hotline: +49 30 40 50 5110
  - LPIC-2 & LPIC-3-certified experts
  - Competent to advise on any issues around Linux, servers, or DMZ
  - Deals with incidents: Down times, Performance issues, Hacking attacks, Data loss
  - Offers strategic advice: Revision, Planning, Consultancy, Configuration advice

Part 1:
Terraform

Robert Sander <r.sander@heinlein-support.de>

Terraform
What does it offer?

- Infrastructure as Code
  - high level configuration syntax as blueprint of data center
- Execution Plans
  - show changes before being applied
- Resource Graph
  - parallelization of execution where possible
- Change Automation
  - minimal interaction

Robert Sander <r.sander@heinlein-support.de>

Terraform
Where does it come from?

- created by HashiCorp
  - also: Vagrant, Packer, Consul, Vault, Nomad, Serf, Sentinel
- initial release July 2014
- current release 0.12.21, February 2020
- written in Go

- https://github.com/hashicorp/terraform
- https://www.terraform.io/

Robert Sander <r.sander@heinlein-support.de>

Terraform
Why do I use it?

- I'm trainer for Ceph
- every student needs the same environment
- 6 - 12 VMs per environment, depending on course content
- up to 9 students plus trainer
- too much manual work

- we have CloudStack
- there's a CloudStack provider for Terraform
- https://www.terraform.io/docs/providers/cloudstack/index.html

Robert Sander <r.sander@heinlein-support.de>

Terraform
What can it do?

- Providers

```
provider "cloudstack" {
  api_url    = …
  api_key    = …
  secret_key = …
}
```

- Resources
  - "create things" depending on provider
- Modules
  - Input Variables
  - contain one or more resource definitions
  - Output Values
- Variables

Terraform
What can it do?

- create network

```
resource "cloudstack_network" "net-01" {
  zone             = zone UUID
  project          = project UUID
  name             = "ceph-tn${var.student}-public"
  display_text     = "Ceph Academy TN ${var.student} Public"
  cidr             = "10.24.4.0/24"
  network_domain   = "ceph.heinlein-akademie.de"
  network_offering = offering UUID
}
```

Terraform
What can it do?

- create instance

```
resource "cloudstack_instance" "vm" {
  zone             = zone UUID
  project          = project UUID
  network_id       = network UUID
  ip_address       = IP UUID
  name             = var.name
  display_name     = "TN ${var.student} ${var.name}"
  service_offering = offering UUID
  template         = template UUID
  keypair          = var.keypair
  expunge          = true
}
```

-

Terraform
What can it do?

- create disk volume

```
resource "cloudstack_disk" "disk-01" {
  name = "tn${var.student}-${cloudstack_instance.vm.name}-01"
  zone                = zone UUID
  project             = project UUID
  disk_offering       = offering UUID
  virtual_machine_id = cloudstack_instance.vm.id
  attach              = true
  device_id           = 4
}
```

Terraform Afterwards

- Terraform does nothing inside the created VMs

- Ansible
- SaltStack
- Puppet
- Chef

Terraform Modules

- sub directories with `main.tf`
- `terraform init` after changing modules
-

```
.
├── ceph-academy.tf
├── cephnode
│   └── main.tf
├── clientnode
│   └── main.tf
├── student
│   └── main.tf
└── variables.tf
```

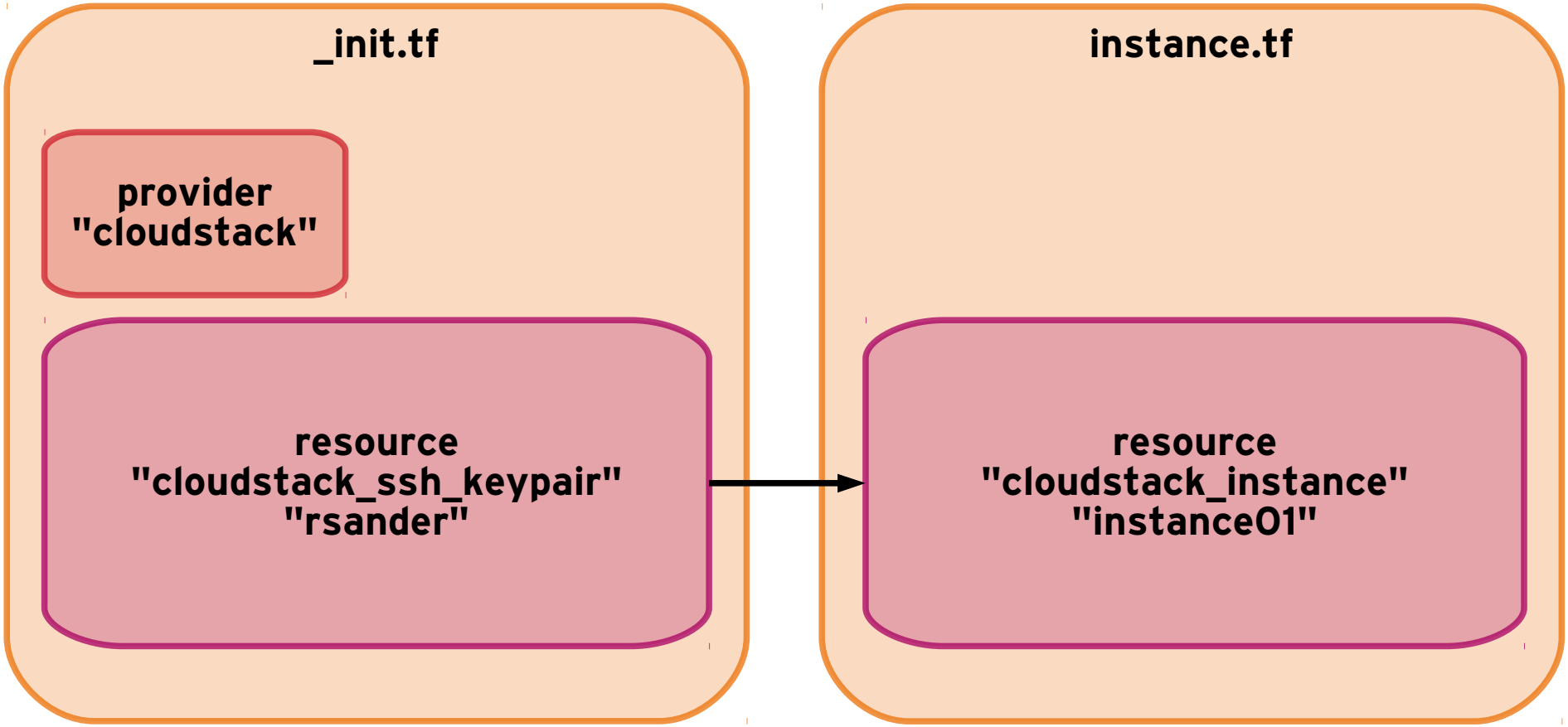
Terraform Projects

- Terraform directory ⇔ CloudStack project

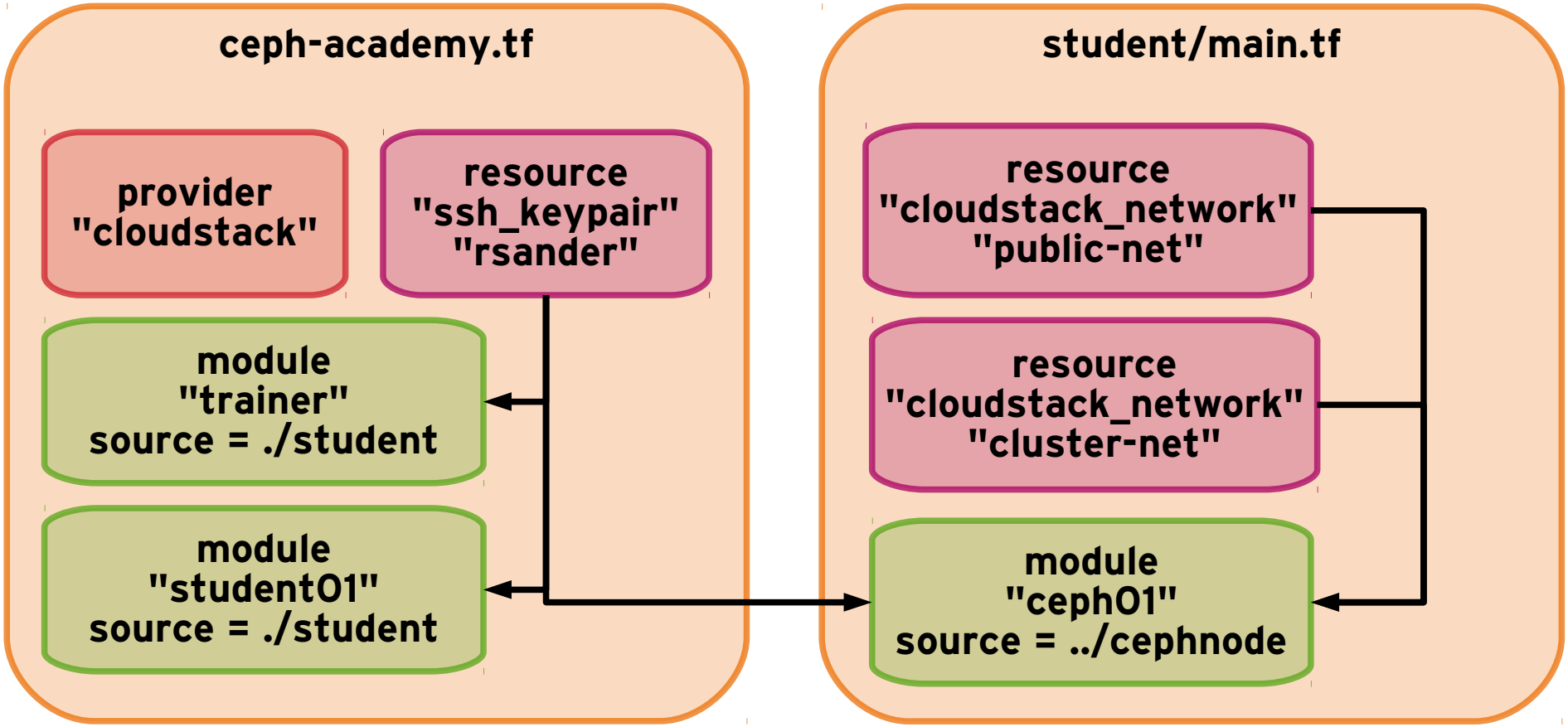
- Terraform reads every `.tf` file in current working directory
- Terraform creates execution plan
- Terraform applies all changes
  - adds new instances
  - removes instances not in config any more
- state is saved in `terraform.tfstate`
  - `terraform refresh`

Part 1:
Demo

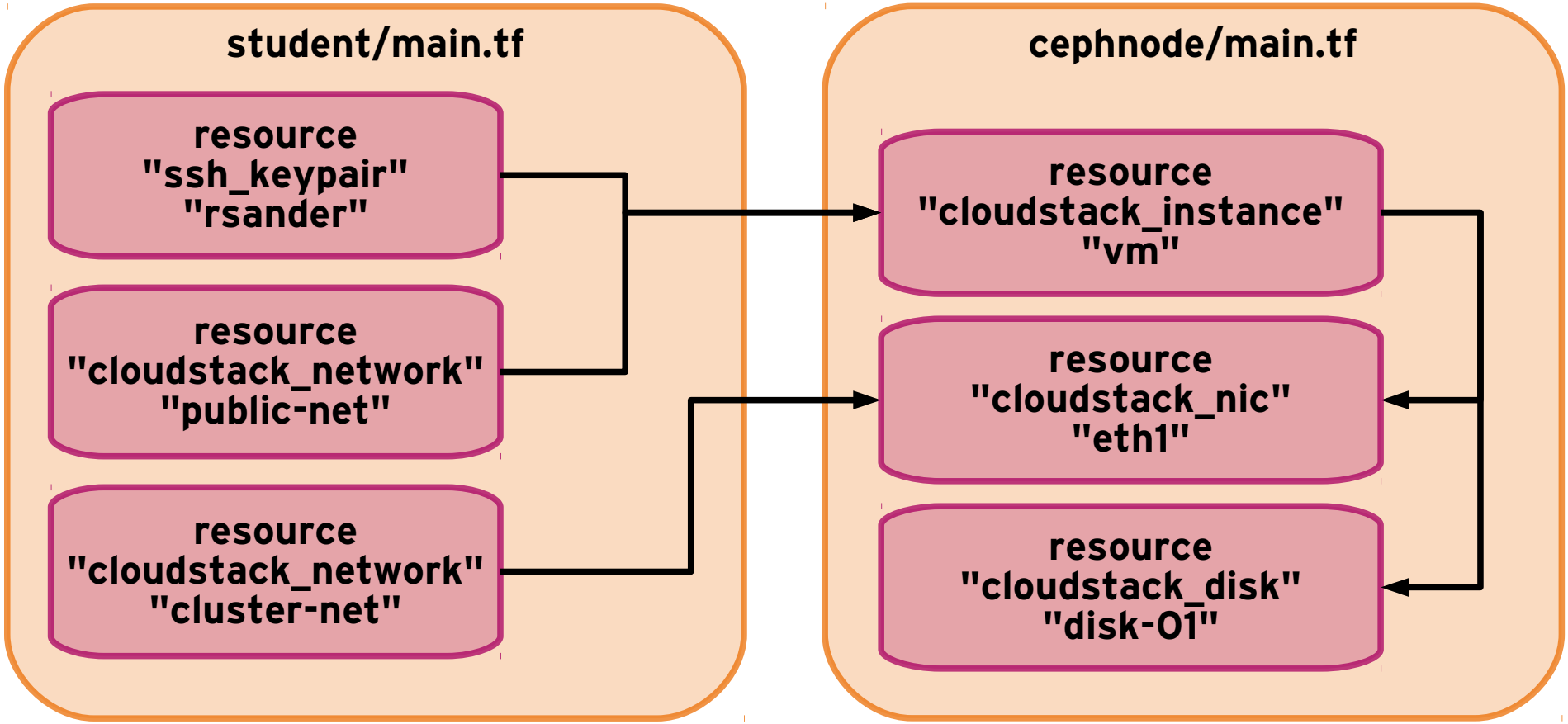
Simple Example



Ceph Academy Example



Ceph Academy Example



- I am always happy to help and look forward to connecting with you.
  - Robert Sander
  - Mail: r.sander@heinlein-support.de
  - Phone: +49 30 40 50 51 43
- If it's urgent:
  - Heinlein Support 24/7 Emergency hotline +49 30 40 50 5110

So far, so good.

Now it is your turn:
Time for questions and discussions!

We are looking for:
Administrators, consultants, and trainers

We are offering:
Interesting projects, appreciation and customer satisfaction, opportunities to work independently, a great team, work-life balance

...and of course: Linux, Linux, Linux...

http://www.heinlein-support.de/jobs

And finally...

- Thank you for your attention!
- We hope you had a productive day and wish you the best of success

Hope to see you again soon!

Robert Sander <r.sander@heinlein-support.de>

Heinlein Support assists with all questions around Linux servers

HEINLEIN AKADEMIE

Professional training by professionals: We convey the top 10% of knowledge: Acquire useful know-how and benefit from our extensive practical experience.

HEINLEIN CONSULTING

The backup for your own Linux administration: LPIC-2-certified professionals help with your CompetenceCall emergencies, also available on SLAs covering 24/7-availability.

HEINLEIN HOSTING

Business hosting tailored to your needs and professional maintenance. Security and availability are our top priorities.