Welcome to

Drift af en infrastruktur med Ansible

Henrik Lund Kramshøj hlk@zencurity.dk

Slides are available as PDF, kramshoej@Github

slide are available as PDF kramshoej@Github

Goal and Agenda: Ansible and more

PatientSky is rolling out new health infrastructure of connected clinics in Norway.

We are very few people running the systems, so we need to automate.

... but automation has other benefits.

Prerequisites: Python, SSH, SSH keys, sudo

Ansible introduction, what is this Ansible

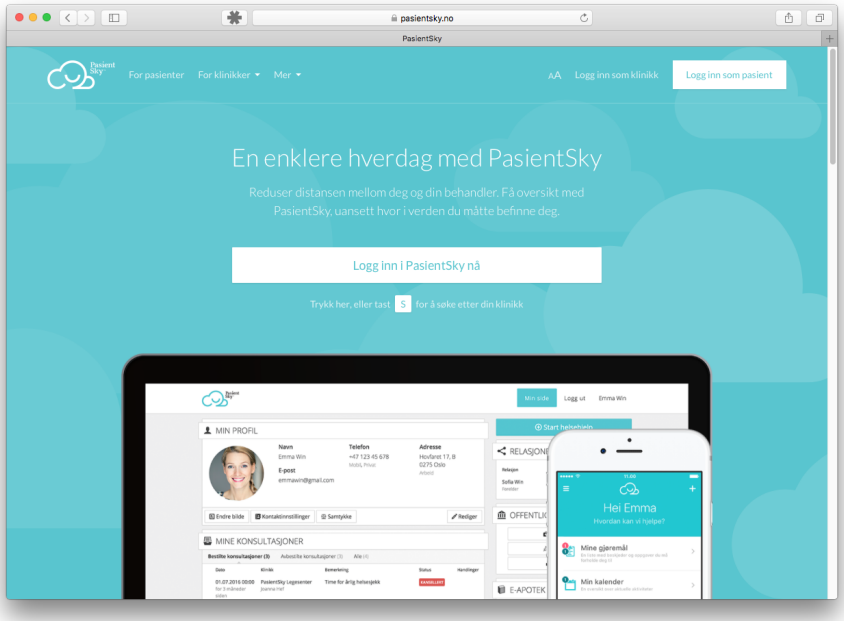
Ansible targets: Linux hosts, ESXi, network devices

Ansible examples, and workshop

Keywords: Ansible, YAML, automating boring stuff

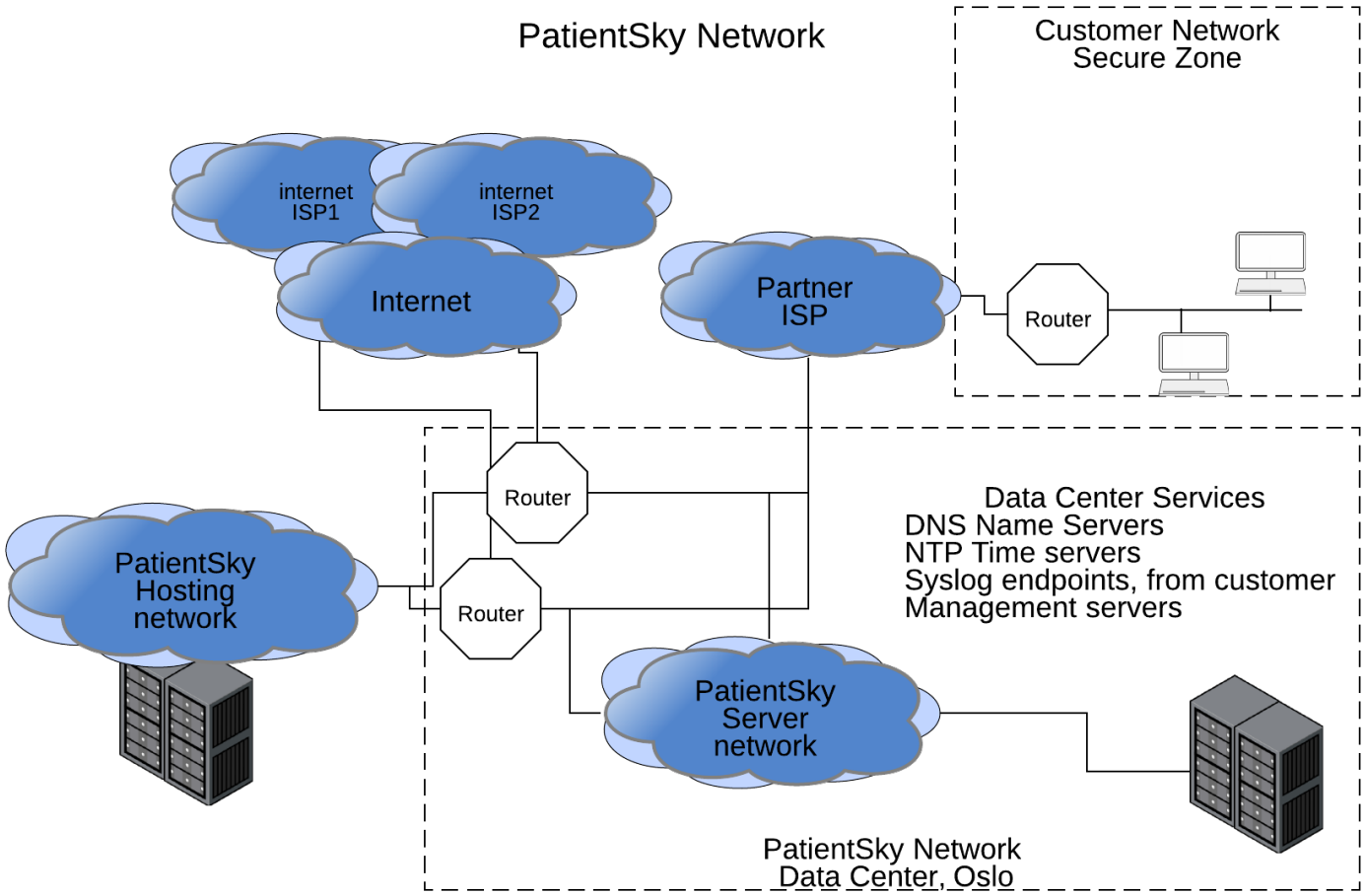
For optimal fun, use your laptop, fetch it in next break!

Pasientsky.no - the environment and services



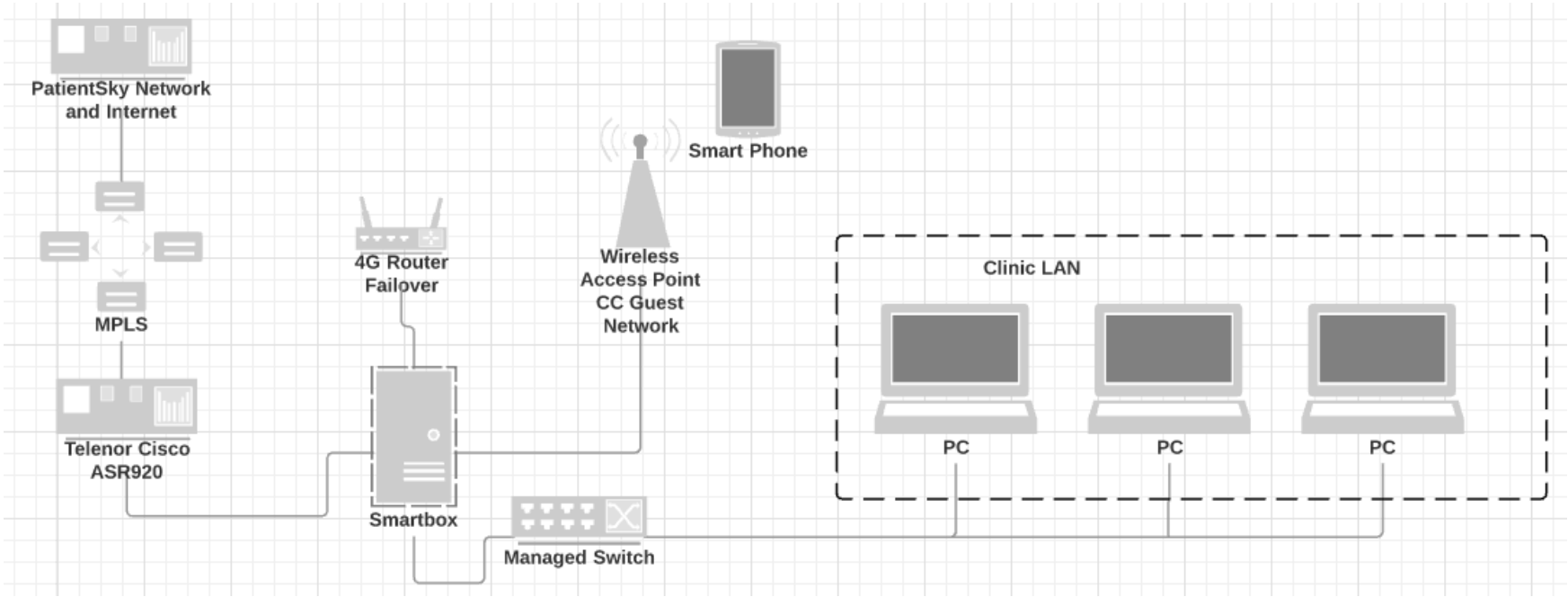
Connected Clinic from PasientSky provides modern and revolutionary solutions meeting the special communication needs in the health sector. A small and smart box provides quick and stable internet connection with integrated telephony and time book.

Overview



Most servers are Linux, percentage is OpenBSD, running on VMware ESXi

OpenBSD CPE: BGP, PF and service daemons



- Soekris Net6501-50 1 Ghz CPU, 1024 Mbyte DDR2-SDRAM, 4 x 1Gbit Ethernet
- OpenBSD operating system
- We install new Smartboxes every week

Important processes and components

- Setup hardware
- Connect cables

- Setup development environment
- Setup staging environment - like development
- Setup production environment - like staging
- Setup firewalls, security, LDAP servers
- Setup other surrounding infrastructure

Top parts hard to automate, bottom easier ☺

What is Ansible

AUTOMATION FOR EVERYONE

Ansible is designed around the way people work and the way people work together.

Ansible has thousands of users, hundreds of customers and over 2,400 community contributors.

750+ Ansible modules

`https://www.ansible.com/`

We have been using Ansible for about 2 years

Warning: we dont really use the roles in Ansible sorry

How Ansible Works: inventory files

```
[all:vars]
ansible_ssh_port=34443

[office]
fw-ps-dk-01 ansible_ssh_host=192.168.1.1 ansible_ssh_port=22
ansible_python_interpreter=/usr/local/bin/python

[infrastructure]
smtp-01     ansible_ssh_host=185.60.160.37 ansible_python_interpreter=/usr/local/bin/python
vpnmon-01   ansible_ssh_host=10.50.22.18
```

- Inventory files specify the hosts we work with
- Linux and OpenBSD servers shown here
- Real inventory for this site with development and staging approx 500 lines

How Ansible Works: ad hoc parallel execution

```
ansible -m ping new-server
ansible -a "date" new-server
ansible -m shell -a "grep a /etc/something" new-server
```

- Running a command on multiple servers is easy now

How Ansible Works: Playbooks

```
- hosts: smartbox-*
  become: yes
  tasks:
  - name: Create a template pf.conf
    template:
      src=pf/pf.conf.j2
      dest=/etc/pf.conf owner=root group=wheel mode=0600
   notify:
      - reload pf
    tags:
      - firewall
      - pf.conf
```

- Almost directly from our ansible repo

How Ansible Works: typical execution

```
ansible-playbook -i hosts.odn1 -K infrastructure-firewalls.yml -t pf.conf --check --diff

ansible-playbook -i hosts.odn1 -K infrastructure-firewalls.yml -t pf.conf

ansible-playbook -i hosts.odn1 -K infrastructure-nagios.yml -t config-only

ansible-playbook -i smartboxes -K create-pf-conf.yml -l smartbox-xxx-01
```

- Pro tip: check before you push out changes to production networks ☺
- Diff will show the changes about to be made

How Ansible Works: atypical execution / gotchas

```
ansible -i ../smartboxes.osl1 --become --ask-become-pass -m shell
-a "pfctl -s rules" -l smartbox01

ansible -i ../smartboxes.osl1 --become --ask-become-pass -m shell
-a "nmap -sP 185.161.1xx.123-124 2> /dev/null| grep done" all
```

- Sometimes you need a trick or persistence
- Ansible moving from *sudo* to *become*
- The normal -K did not work, but the above does for ad hoc commands

Stop: discussion benefits of Ansible

Do we even need to run the same command on multiple servers?

What are the benefits of Ansible?

- Central configuration management - git repo
- Same playbook - different inventory file, what happens
-

Up and running with Ansible

Prequisites for Ansible:

- python language - Ansible uses this
- ssh keys - remote login without passwords
- Sudo - allow regular users to do superuser tasks
- Recommended tool: `ssh-copy-id` for getting your key on new server
- Recommended Change: `sshd_config` - no passwords allowed, no bruteforce
- Recommended to use: jump hosts/ProxyCommand in `ssh_config`

Install python on servers

- Ubuntu server: `apt install python`
- OpenBSD: `pkg_add python`
  Requires `PKG_PATH` set, see next slide

OpenBSD python

/root/.profile

```
PKG_PATH=ftp://mirror.one.com/pub/OpenBSD/`uname -r`/packages/`uname -m`
PKG_PATH=https://stable.mtier.org/updates/$(uname -r)/$(arch -s):$PKG_PATH
export PKG_PATH
```

-
-
-
-

-
-
-
-

Conclusion

Automation is cool - use it