

## MAKALAH SEMINAR

# OTOMASI INSTALASI DAN KONFIGURASI KOMPUTER PADA LAB ILMU KOMPUTER IPB DENGAN SALTSTACK

## *AUTOMATION OF COMPUTER INSTALLATION AND CONFIGURATION ON IPB COMPUTER SCIENCE LAB WITH SALTSTACK*

Moh. Azis<sup>1\*</sup>, Auriza Rahmad Akbar<sup>1</sup>

<sup>1</sup>*Departemen Ilmu Komputer, IPB, Bogor, Jawa Barat, 16680, Indonesia,  
Email: aziscs20@gmail.com*

### Abstrak

Penggunaan komputer semakin meningkat pada bidang pendidikan. Sekolah dari tingkat SMP dituntut untuk memiliki lab komputer bagi siswanya. Dalam pembangunan lab komputer, konfigurasi tiap komputer harus seragam. Selama ini, instalasi dilakukan secara manual, sehingga membuang tenaga dan waktu untuk menjamin keseragaman konfigurasi komputer. Untuk skala besar, diperlukan sebuah otomasi sehingga instalasi dapat dilakukan secara efektif dan efisien. Penelitian ini akan menerapkan manajemen instalasi dan konfigurasi menggunakan SaltStack. Metode yang digunakan yaitu identifikasi kebutuhan perangkat lunak lab, penerapan otomasi dengan SaltStack, validasi hasil instalasi, dan konfigurasi, dan pengujian kinerja pada mesin virtual. Pengujian kinerja dilakukan dengan membandingkan waktu instalasi dengan *shell script* (semi-otomatis) dan dengan SaltStack. SaltStack berhasil menginstal perangkat lunak lab pada 4 komputer sekaligus dengan waktu instalasi sekitar 70 menit. Kinerjanya hanya 2% lebih cepat daripada *shell script*, namun SaltStack menyediakan *log* instalasi yang akan mempermudah jika terjadi masalah.

Kata kunci: lab komputer, manajemen instalasi dan konfigurasi, otomasi, SaltStack

### Abstract

*The use of computers is increasing in education. Schools of junior high school are required to have a computer lab for their students. In computer lab development, each computer configuration must be uniform. All this time, the installation is done manually, thereby wasting power and time to ensure uniformity of computer configuration. For large scale, it needs an automation so that installation can be done effectively and efficiently. This research will implement installation and configuration management using SaltStack. The method used is the identification of the needs of the software lab, the application of automation with SaltStack, the validation of the installation results, configuration and performance testing on the virtual machine. Performance testing is done by comparing installation time with shell script (semi-automatic) and with SaltStack. SaltStack manages to install lab software on 4 computers at once with installation time of about 70 minutes. It's performance is only 2% faster than the shell script, but SaltStack provides an installation log that will make it easier if a problem occurs.*

*Keywords: computer lab, installation and configuration management, automation, SaltStack.*

## PENDAHULUAN

### Latar Belakang

Manajemen instalasi sistem adalah sistem kontrol berbasis komputer untuk memantau dan mengelola suatu perangkat, dalam proses menginstal program komputer seperti aplikasi, driver atau perangkat lunak lainnya (Ambasta *et al.* 2016; Blount *et al.* 2017). *Deployment*

---

\* Penulis korespondensi

manajemen instalasi merupakan penyebaran instalasi dan konfigurasi terhadap perangkat lunak komputer sehingga memiliki konfigurasi secara seragam.

Manajemen instalasi dan pengolahan aplikasi komputer masih dilakukan secara manual sehingga membutuhkan waktu yang lama dan sumber daya yang banyak. Masalah yang sering terjadi tanpa adanya *deployment* sulit dalam melakukan pemantauan, pengelolaan dan pembaharuan terhadap perangkat lunak komputer. Manajemen instalasi dan konfigurasi yang banyak dilakukan dengan cara melakukan general hardware-oriented system transfer (ghost) dan cloning, ghost merupakan teknik manajemen instalasi dan konfigurasi dengan cara melakukan instalasi dan konfigurasi terhadap satu komputer, dari hasil instalasi dan konfigurasi dilakukan proses kloning disk. Proses kloning disk sering mengalami masalah yaitu driver dan perangkat lunak komputer sering mengalami kerusakan dan tidak berjalan secara normal dan harus dilakukan terhadap komputer yang memiliki spesifikasi perangkat yang sama. Jeffrey *et al.* (2012) menyatakan banyak aplikasi modern membutuhkan instalasi yang saling tergantung, terdistribusi dan diciptakan secara mandiri, sehingga aplikasi dapat memberikan fungsionalitas secara signifikan dan terukur. Menurut Fischer *et al.* (2012), manajemen instalasi dan konfigurasi pada komputer yang dilakukan secara manual cenderung mengalami kesalahan akibat sering terjadi pembaharuan aplikasi secara berkala dan beberapa aplikasi yang mengalami kerusakan. Hardion *et al.* (2013) menyatakan bahwa adanya perbedaan konfigurasi pada setiap komputer cenderung memiliki resiko adanya perubahan perilaku pada perangkat lunak.

Beberapa penelitian telah dilakukan mengenai penyebaran instalasi (*deployment*) sistem dan konfigurasi, Fischer *et al.* (2012) melakukan penelitian mengenai Engage (sebuah sistem manajemen *deployment*) dengan cara melakukan percobaan secara eksplisit dan memodelkan konfigurasi metadata dan interkomponen yang bersifat dependensi. Engage memungkinkan pengecekan konfigurasi aplikasi secara otomatis sehingga mengurangi proses konfigurasi secara manual yang dikerjakan secara intensif. Percobaan yang lainnya dilakukan oleh Luo *et al.* (2015) tentang sistem *deployment* untuk aplikasi *cloud*.

Lab komputer saat ini masih bersifat penerapan dengan menggunakan *shell script*. Pada penelitian ini akan menerapkan otomatisasi manajemen instalasi dan konfigurasi pada lab komputer dengan berbasis SaltStack. SaltStack merupakan sebuah perangkat lunak pengelolaan infrastruktur. SaltStack dapat digunakan untuk eksekusi remote untuk infrastruktur apapun dan manajemen konfigurasi untuk stack aplikasi apapun (Myers 2016; SaltStack 2017).

### **Perumusan Masalah**

Rumusan masalah pada penelitian ini adalah bagaimana *deployment* berbasis SaltStack dapat diterapkan dalam manajemen instalasi dan konfigurasi lab komputer Ilmu Komputer IPB.

### **Tujuan Penelitian**

Tujuan penelitian ini yaitu menerapkan *deployment* berbasis SaltStack dalam otomatisasi manajemen instalasi dan konfigurasi pada lab komputer Ilmu Komputer IPB.

### **Manfaat Penelitian**

Manfaat penelitian ini yaitu mempermudah dalam manajemen instalasi dan konfigurasi pada lab komputer untuk mempermudah dan memperringkas pekerjaan yang bersifat repetitif.

### **Ruang Lingkup**

Ruang lingkup penelitian ini, yaitu

- 1 Otomatisasi instalasi dan konfigurasi dilakukan dengan menggunakan SaltStack
- 2 Pengujian Instalasi dan konfigurasi menggunakan mesin virtual
- 3 Daftar perangkat lunak dan konfigurasi disesuaikan dengan kebutuhan di lab komputer Ilmu Komputer IPB

## **METODE**

### **Identifikasi Masalah**

Proses instalasi dan konfigurasi sistem pada komputer jika dilakukan pada sebuah komputer masih dilakukan secara manual. Misalnya, pembangunan lab komputer yang memerlukan konfigurasi yang sama. Jika proses instalasi dan konfigurasi dilakukan secara manual dapat membutuhkan waktu yang lama dan cenderung mengalami kesalahan.

### **Identifikasi Kebutuhan**

Identifikasi kebutuhan yaitu pada tahap ini mengidentifikasi sistem operasi yang akan digunakan. Identifikasi kebutuhan yang kedua yaitu identifikasi jenis aplikasi yang akan digunakan di lab komputer Ilmu Komputer IPB sesuai mata kuliah yang diselenggarakan.

### **Implementasi**

Instalasi sistem operasi Ubuntu pada komputer dan menset alamat IP statis, meinstal SSH, dan mengarahkan repositori Ubuntu ke server lokal. Aplikasi VM diinstal pada sistem operasi dan diisi beberapa sistem operasi di dalam VM sebagai komputer klien. Daftar aplikasi berdasarkan Tabel 1 dan 2 akan diinstal.

### **Pengujian di Mesin Virtual**

Pada mesin virtual dilakukan instalasi sistem operasi Xubuntu sebagai master dan klien. Jumlah mesin virtual yang akan dibuat yaitu empat mesin virtual. Pada sistem operasi master diinstal SSH, diatur alamat IP statis, dan mengarahkan repositori Ubuntu ke server lokal. Setelah dilakukan instalasi pada komputer master, dilakukan proses instalasi dan konfigurasi perangkat lunak sesuai Tabel 1 dan 2.

### **Analisis Hasil dan Kinerja**

Analisis hasil dan kinerja dari hasil pengujian yang dilakukan pada pengujian pada mesin virtual. Parameter uji yang akan digunakan yaitu hasil dan kinerja. Pada parameter uji hasil yaitu dilakukan pengecekan terhadap perangkat lunak yang diinstal dari konfigurasi dan parameter uji kinerja yaitu berupa waktu instalasi pada komputer klien dengan jumlah 1, 2, 3 dan 4.

### **Peralatan Penelitian**

Spesifikasi perangkat keras dan perangkat lunak yang digunakan untuk penelitian ini adalah sebagai berikut:

1. Perangkat keras yang digunakan terdiri dari dua perangkat keras yang digunakan dalam pengujian di mesin virtual sebagai berikut:
  - Komputer untuk pengujian mesin virtual
    - Prosesor: Intel Core i7-4770 4C/8T 3.40 GHz
    - RAM: DDR3 32 GiB
    - HDD: 1 TiB
    - NIC: Gigabit Ethernet
  - Spesifikasi mesin virtual
    - Prosesor: 1 CPU
    - HDD: 100 GiB
    - RAM: 4 GiB
    - NIC: *bridged*
2. Perangkat lunak yang digunakan
  - Sistem operasi Xubuntu 16.04
  - VirtualBox 5.0.40
  - SaltStack 2018.3

## HASIL DAN PEMBAHASAN

### Identifikasi Kebutuhan

Kebutuhan dalam manajemen instalasi dan konfigurasi sistem menggunakan SaltStack adalah sistem operasi Xubuntu dengan melakukan pengaturan alamat IP pada komputer server, perangkat lunak dasar dan jenis perangkat lunak yang dipakai lab komputer IPB seperti yang tertera pada Tabel 1 dan 2.

Tabel 1 Daftar perangkat lunak masing - masing mata kuliah

| Mata kuliah  | Perangkat lunak  |
|--------------|--|
| Penkom       | Office   |
| Penkom Mulok | geany, texlive, pandoc, pandoc-citeproc  |
| Algor        | codeblocks, geany  |
| Alpro        | python3, PyCharmEdu  |
| Dasprog      | Codeblocks   |
| Radig        | Logisim, Fritzing, iverilog, gtkwave, eagle  |
| Basprog      | gprolog, racket, g++, jdk8   |
| Strukdat     | codeblocks, geany  |
| OAK          | yasm, gdb, ddd   |
| Basdat       | postgresql, apache2, php, pgadmin3, Access   |
| RPL          | postgresql, apache2, php, sublime-text, Postman  |
| PMK          | R  |
| Metkuan      | R  |
| Komnum       | R  |
| AI           | Python   |
| Grafkom      | codeblocks, libglfw3-dev, unity-editor, android-studio   |
| SO           | htop, tree, tesseract-ocr  |
| IMK          | GravitDesigner, inkscape, Browser, Word Processor  |
| PCD          | codeblocks, opencv   |
| PSBO         | Monodevelop  |
| Komdat       | PacketTracer, virtualbox, nmap, traceroute, whois, wireshark, bmon                                       |
| Damin        | R, weka  |
| Siscerdas    | R  |
| Bioinfor     | R, python, ugene, clustal, velvet, soapdenovo, bowtie, samtools, plink, MetaSim, TASSEL, MEGAN, I-TASSER |
| PTG          | qgis, GeoServer, postgresql, postgis, libjs-openlayers   |
| KI           | python, pycrypto   |
| NLP          | python, Matlab   |
| Paralel      | MPI  |

Tabel 2 Daftar perangkat lunak dasar yang dibutuhkan

| Jenis        | Perangkat lunak                |
|--------------|--------------------------------|
| Virtualisasi | Virtualbox                     |
| Kompresi     | p7zip, unrar                   |
| Office       | libreoffice-impress            |
| Grafis       | dia, gimp, inkscape, graphviz  |
| Multimedia   | mpv, audacity, vokoscreen      |
| Adblock      | xul-ext-ublock-origin          |
| CLI app      | htop, tmux, tree, w3m, ntpdate |
| Ekstra       | xubuntu-restricted-extras      |
| Editor       | geany, geany-plugins           |
| Programing   | build-essential, git           |

### Implementasi Pengujian di Mesin Virtual

Empat mesin virtual komputer yang diinstal sistem operasi Xubuntu dengan *hostname* yang dibuat pada setiap mesin yaitu lab-71, lab-72, lab-73 dan lab-74. Pada mesin virtual lab-71 menjadi SaltMaster sekaligus menjadi SaltMinion, melakukan pemasangan perangkat lunak SaltMaster dan SaltMinion, setiap komputer dilakukan pengaturan IP statis dan mengubah repositori Xubuntu menjadi repositori lokal seperti berikut ini.

```
# set IP statis
eth="$(ls /sys/class/net | grep en)"
host="$(hostname | tr -d a-z-)"
net="172.18.12"
cat << EOF | sudo tee -a /etc/network/interfaces
auto $eth
iface $eth inet static
    address $net.$host
    netmask 255.255.255.0
    gateway $net.254
    dns-nameservers 172.17.5.14 172.17.5.21
EOF
sudo service networking restart

# SaltStack 2018.3 (latest)
wget -O - https://repo.saltstack.com/apt/ubuntu/16.04/amd64/latest/SALTSTACK-GPG-KEY.pub |
    sudo apt-key add -

# set repository
sudo tee /etc/apt/sources.list << EOF
deb http://172.18.12.13/ubuntu/ xenial main universe multiverse restricted
deb http://172.18.12.13/ubuntu/ xenial-security main universe multiverse restricted
deb http://172.18.12.13/ubuntu/ xenial-updates main universe multiverse restricted
deb http://repo.saltstack.com/apt/ubuntu/16.04/amd64/latest xenial main
EOF
```

Pada SaltMaster dilakukan pengaturan konfigurasi yang terdapat pada *file* `/etc/salt/master` dengan melakukan perubahan pada bagian *interface* dan *publish\_port*.

```
#### /etc/salt/master
## Primary configuration settings
interface: 172.18.12.71
#ipv6: False
publish_port: 4505
## File Server settings
file_roots:
    base:
        - /srv/salt
```

Pada SaltMinion, konfigurasi pada file `/etc/salt/minion` sebagai berikut.

```
#### /etc/salt/minion
## Primary configuration settings
master: 172.18.12.71
id: minion.0
## File Directory Settings
file_roots:
  base:
    - /srv/salt
```

Pada bagian SaltMinion diperlukan persetujuan dari SaltMaster dengan menggunakan modul `salt-key -A` untuk menerima semua SaltMinion yang ada, sedangkan untuk menerima sebagian SaltMinion yaitu menggunakan modul `salt-key -a` `IDSaltMinion`. Cara penulisan modul pada SaltStack seperti tertera pada Gambar 1. Modul yang digunakan untuk melakukan pemasangan aplikasi pada Tabel 1 dan 2 yaitu `cmd`, `pkg` `user`, `file` dan modul fungsi yang digunakan yaitu `state.apply`.

```
salt '*' test.rand_sleep 120
```

target                      module.function                      arguments

Gambar 1 Cara penulisan *remote execution* SaltStack

### Modul `cmd`

Pada penelitian ini modul `cmd` yang digunakan ada dua yaitu fungsi `cmd.run` dan `cmd.script`. Penerapan fungsi `cmd.run` dan `cmd.script` seperti berikut ini.

```
saltstack_downloadapp:
cmd.run:
  - name: wget
    http://172.18.12.13/lab/jdk-8u161-linux-x64.tar.gz
    http://172.18.12.13/lab/PacketTracer711_64bit_linux.tar.gz
    http://172.18.12.13/lab/rstudio-xenial-1.1.414-amd64.deb
    http://...
saltstack_script0:
cmd.script:
  - source: /srv/salt/opt.sh
```

### Modul `pkg`

Pada penelitian ini modul `pkg` yang digunakan untuk instalasi program dan konfigurasi adalah fungsi `pkgs.installed`. Penerapan fungsi `pkgs.installed` seperti berikut ini.

```
saltstack_pkginstal:
pkg:
  - installed
  - pkgs:
    - virtualbox
    - p7zip
    - unrar
    - libreoffice-impress
    - libreoffice-base
    - dia
    - ...
```

### Modul `file`

Pada penelitian ini modul `file` yang digunakan yaitu fungsi `file.managed` dan `file.absent`. Penerapan fungsi `file.absent` seperti berikut ini.

```
file_delete_eula.txt:
  file.absent:
    - names:
      - /srv/salt/myapp/eula.txt
    ...
```

## Modul user

Pada Penelitian ini modul user yang digunakan yaitu fungsi `user.present`. Penerapan fungsi `user.present` seperti berikut ini.

```
saltstack_adduser:
  user.present:
    - name: student
    - password: $6$Kh9rNhbm$GjHMGtoH7rCt/UCHqnsC0gB1BKJvEd0d7sW7yGiTJXSbZJHWGdbWRXRfmBg00...
```

## Analisis Hasil dan Kinerja

SaltStack dapat menghasilkan hasil kinerja yang lebih efektif dan efisien dibandingkan dengan proses instalasi semi otomatis dengan *shell script*. Tabel 4 merupakan hasil perhitungan waktu pada pengujian instalasi perangkat lunak pada Tabel 1 dan 2.

| Tabel 4 Hasil waktu pengujian instalasi program dengan 4 komputer client |                |                   |
|--|----------------|-------------------|
| Iterasi  | Manual (detik) | SaltStack (detik) |
| 1  | 4216           | 4128              |
| 2  | 4389           | 4306              |
| 3  | 4323           | 4238              |
| Rataan   | 4309           | 4223              |

Berdasarkan pengujian yang dilakukan dengan jumlah klien 4 komputer pada iterasi pertama waktu yang dibutuhkan untuk instalasi secara semiotomatis dan SaltStack masing-masing sebesar 70.3 menit dan 68.8 menit. Pada iterasi kedua waktu yang dibutuhkan secara semiotomatis adalah 73.2 menit dan menggunakan SaltStack adalah 71.8 menit. Pada iterasi ketiga membutuhkan waktu 72.0 menit untuk instalasi secara semiotomatis dan 70.4 untuk instalasi menggunakan SaltStack. Dari data diatas instalasi menggunakan saltstack lebih efektif dan efisien dibandingkan semiotomatis. Hal ini karena pada SaltStack memiliki dasar penulisan yang baku dalam menjalankan modul yang akan digunakan. SaltStack dapat digunakan sebagai perangkat lunak yang dapat mempermudah untuk melakukan perawatan secara berkala terhadap berbagai program yang terpasang pada komputer. SaltStack lebih efektif dalam menemukan masalah yang terjadi pada perangkat lunak yang mengalami masalah dalam proses instalasi dan konfigurasi. Hal ini karena SaltStack memiliki sebuah ringkasan terhadap modul-modul yang dijalankan seperti yang tertera pada Gambar 2.

## SIMPULAN

SaltStack berhasil melakukan manajemen instalasi dan konfigurasi perangkat lunak pada beberapa komputer sekaligus dengan hasil yang seragam. SaltStack menyediakan modul beserta dokumentasi lengkap mengenai cara penggunaannya. Hasil kinerja menunjukkan SaltStack lebih cepat hanya 2% daripada menggunakan *shell script* (semi-otomatis). Namun bedanya, SaltStack bersifat *full* otomatis sehingga teknisi tidak perlu mengawasi proses instalasi yang panjang hingga selesai. Selain itu, SaltStack menyediakan *log* hasil instalasi yang memudahkan teknisi untuk menemukan sumber masalah jika terjadi kesalahan. Dengan demikian, SaltStack cocok digunakan untuk *deployment* infrastruktur komputer, khususnya pada pembangunan lab komputer.

```

minion.1:
-----
      ID: saltstack_user
Function: user.present
      Name: student
      Result: True
      Comment: New user student created
      Started: 21:36:55.242868
      Duration: 409.938 ms
      Changes:
              -----
              fullname:
              gid:
                  1001
              groups:
                  - student
              home:
                  /home/student
-----
      ID: saltstack_pkg
Function: pkg.installed
      Result: True
      Comment: 77 targeted packages were installed/updated.
                The following packages were already installed: gdb, git
      Started: 21:36:57.351360
      Duration: 719553.641 ms
      Changes:
              -----
              aglfn:
              -----
              new:
                  1.7-3
              old:

```

Gambar 2 Ringkasan modul SaltStack

## DAFTAR PUSTAKA

- Ambasta RA, Ganesh B, Kulkarni AA, Wani US. 2016. Computer program installation across multiple memories. *US Patent*. US9317420 B2.
- Barret D, Silverman R, Byrnes R. 2005. *SSH, the Secure Shell: The Definitive Guide*. California (US): O'Reilly.
- Blount ML, Kephart JO, Lenchner J, Mohamed II. 2017. Sensor installation in a building management system. *US Patent*. US9575477 B2.
- Fischer J, Majumdar R, Esmaeilsabzali S. 2012. Engage: a deployment management system. *SIGPLAN*. 47(6): 263-274.
- Hardion V, Spruce DP, Lindberg M, Otero AM, Simon JL, Jamroz JJ. 2013. *Configuration Management Of The Control System*. San Francisco (USA). ICALEPCS: 1114-1117.
- Kurniawan A, Palit HN, Adjarwirawan J. 2016. Eksplorasi pemanfaatan Docker untuk mempermudah pengelolaan instalasi komputer di laboratorium komputer Teknik Informatika Universitas Kristen Petra. *Jurnal Infra*. 4(2): 130-135.
- Luo R, Ye W, Zhang S. 2015. Towards a Deployment System for Cloud Applications. Di dalam: *27 th Intenational Conference on Software Engineering and Knowledge Engineering*. 2015 Jul 6-8; Pittsburgh US. Hal: 127-127.
- Myers C. 2016. *Learning SaltStack: Build, Manage, and Secure Your Infrastructure by Utilizing the Power of SaltStack*. Birmingham (GB): Packt.
- Smith JE, Nair R. 2005. *Virtual Machines: Versatile Platforms for Systems and Processes*. San Francisco (US): Elsevier.