



Proyek Uji Lomba Kompetensi Siswa Nasional 2026

SMA/MA/SMK/MAK/Sederajat



Cabang Ajang

**Teknologi Informasi Sistem Administrasi Jaringan
(IT Network System Administration)**

ACTUAL TEST PROJECT
MODUL D - LINUX ENVIRONMENT

IT NETWORK SYSTEMS ADMINISTRATION

**LOMBA KOMPETENSI SISWA DIKMEN
TINGKAT NASIONAL 2026**

Introduction

You have been appointed as the system administrator for the company which utilizes Linux as its primary technology for service and resource solutions. Your role is to ensure all services are properly configured, integrated, and operational within the given time constraints. Attention to detail and time management are critical to your success in this assignment

On every host, there is one network adapter called **Management**. This network adapter will be used for assessment or marking (**DO NOT CHANGE OR MODIFY IT!**). Also local user called **sysadmin** already configured as **sudoers**

Outline of this test project, you must deliver public resources using domain **itnsa.id**, then for internal management it uses **itnsa.local** as the local domain name

There is an external client called **budi-client** (*Hostname & IP address already preconfigured*) that will be used for public services testing provided by **itnsa.id**, also to simulate remote workers that connect VPN to access internal network. Also inside **budi-client**, there is preinstalled **Zealdocs** and **Visual Studio Code**

On each server there is an agent for marking that runs on port **9000**. **DO NOT DISABLE OR MODIFY IT!**

Credential Information

Username: root / sysadmin / budi (budi-client only)

Password: Skill39@2026

Timezone: Asia/Jakarta

Description of project and tasks

Basic Configuration

Configure **hostname**, **FQDN** and **IP address** on all servers of itnsa.id refer to the information table and set the timezone to **(UTC+07:00) Bangkok, Hanoi, Jakarta**.

Make sure every server inside INT & DMZ uses **int-mail.itnsa.id (192.168.1.1)** as the primary nameserver.

DNS Service

int-mail.itnsa.id

1. Configure this server as primary name server for **itnsa.local**
2. Create the following Forward DNS records in the **itnsa.local** zone, save the zone file as **/etc/bind/db.itnsa.local**:

Type	Record Name	Value / Target
A	int-mail	192.168.1.1
A	ha-prx01	192.168.2.10
A	ha-prx02	192.168.2.20
A	web01	192.168.2.101
A	web02	192.168.2.102
A	vrrp	192.168.2.100
A	fw	192.168.1.254
CNAME	web	vrrp.itnsa.local.
CNAME	www	vrrp.itnsa.local.
CNAME	pub-dns	vrrp.itnsa.local.

2. Create the following Static Reverse DNS (PTR) records, save the zone file as **/etc/bind/db.192**:

Record Name	IP Address
int-mail.itnsa.local	192.168.1.1
ha-prx01.itnsa.local	192.168.2.10
ha-prx02.itnsa.local	192.168.2.20
web01.itnsa.local	192.168.2.101
web02.itnsa.local	192.168.2.102

3. Configure **192.168.2.10** and **192.168.2.20** as the DNS forwarders.

ha-prx01.itnsa.id

1. Configure this server as primary name server for **itnsa.id** using bind9
2. Create the following Forward DNS records in the **itnsa.id** zone

Type	Record Name	Value / Target
NS	@	ha-prx01.itnsa.id.
NS	@	ha-prx02.itnsa.id.
MX	@	39 mail.itnsa.id.
A	ha-prx01	192.168.2.10
A	ha-prx02	192.168.2.20
A	pub	138.24.11.24
A	vpn	138.24.11.24
A	mail	138.24.11.24
CNAME	app	pub.itnsa.id.
CNAME	www	pub.itnsa.id.

3. Perform DNS transfer to secondary name server and make enable DNS notify so that zone will be transferred automatically if there is any update on DNS zone
4. Save bind9 log on file called **/var/log/dns.log**

ha-prx02.itnsa.id

1. Configure this server as secondary name server for **itnsa.id** using bind9
2. Make sure this server receive DNS zone of **itnsa.id** from **ha-prx01.itnsa.id**
3. Save bind9 log on file called **/var/log/dns.log**

Root Certificate Authority

int-mail.itnsa.id

1. Configure Root Certification Authority on this server with specification refer to the list below:
 - Use directory **/etc/RootCA** as the directory
 - Set certificate subject to **CN=ITNSA-ID-ROOT-CA**
 - Certificate must be valid for **15 years**
 - Save the root CA certificate as **/etc/RootCA/RootCA.crt**
 - You can add others certificate options too
2. Distribute the root certificate to entire host and servers

Subordinate Certificate Authority

fw.itnsa.id

1. Configure Subordinate Certification Authority on this server with specification refer to the list below:
 - Use directory **/etc/SubCA** as the directory
 - Set certificate subject to **CN=ITNSA-CA**
 - Certificate must be valid for **5 years**
 - This certificate must be issued and signed by **ITNSA-ID-ROOT-CA**
 - Save the CA certificate as **/etc/SubCA/ITNSA-CA.crt**
 - You can add others certificate options too
2. Distribute the root certificate to entire host and servers

Storage & Partition

int-mail.itnsa.id

1. Install LVM on this server and create new partition **/dev/sda2** with maximum size
 - Convert created partition (/dev/sda2) into physical volume
 - Create volume group called **ROOTVG** then add created partition as the member
 - Create logical volume refer to the list below:
 - **data** with size **1GB**
 - **backup** with size **3GB**
2. Format created logical volume using **ext4** filesystem then mount it into **/stg/<LVNAME>** and make sure it persist across reboot

File Service

int-mail.itnsa.id

1. Install NFS service and configure this server as NFS server
2. Export directory **/stg/data** and **/stg/backup** and make sure only client from **192.168.2.0/24** can mount this directory

Note: If you can't create valid logical volume, you can create /stg/data and /stg/backup manually but of course you will not get any point related to the Storage & Partition

web01.itnsa.id & web02.itnsa.id

1. Install NFS client service so that this server can mount NFS share provided by int-mail
2. Mount the **/stg/data** share into local directory **/data** and make sure it persists across reboot

Web Service

web01.itnsa.id & web02.itnsa.id

1. Install any web server of your choice as long it's listen on port **80**
2. Install any PHP interpreter of your choice as long it can be used to serve **.php** file
3. Create virtual host with specification refer to the list below:
4. Create and configure websites according to the specifications of table below:

Website Name	Path	Default Index	Index File Content
www.itnsa.id	/data/www/public	index.html	This is itnsa.id
app.itnsa.id	/data/www/app	index.php	<?php phpinfo(); ?>

- For the .php file, make sure it shows valid information of PHP interpreter
- Add custom header called **X-Served-By** with value hostname of server, example:

```
X-Hostname: web01
```

LDAP Service

int-mail.itnsa.id

- Install and configure LDAP service using OpenLDAP and use domain **itnsa.id**
- Set default admin password to **Skill39@2026**
- Create LDAP object refer to table below:

Distinguished Name	Object Class
ou=Users,dc=itnsa,dc=id	Organizational Unit
ou=Groups,dc=itnsa,dc=id	Organizational Unit
cn=Tech,ou=Groups,dc=itnsa,dc=id	Posix Group
cn=Worker,ou=Groups,dc=itnsa,dc=id	Posix Group
uid=tech01,ou=Users,dc=itnsa,dc=id	Posix Account
uid=tech02,ou=Users,dc=itnsa,dc=id	Posix Account
uid=worker01,ou=Users,dc=itnsa,dc=id	Posix Account
uid=worker02,ou=Users,dc=itnsa,dc=id	Posix Account

- For Posix Account LDAP object, set default password into **Skill39@2026** and make sure it can be used for authentication

Proxy and Failover

ha-prx01.itnsa.id & ha-prx02.itnsa.id

1. Install and configure HAProxy and Keepalived on both server
2. Setup Virtual IP using Keepalived with specification refer to the list below:
 - Use IP 192.168.2.100/24 as the virtual IP
 - Set ha-prx01 as the MASTER with priority 199
 - Set ha-prx02 as the BACKUP with priority 155
3. Configure HAProxy with specification refer to the list below:
 - Distribute HTTP traffic to two web server (web01 & web02)
 - All HTTP requests must be redirected to HTTPS
 - Use a certificate issued by **ITNSA-CA**. You can use any certificate subject common name as long it is valid, even wildcard too
 - Enable health check to the backend servers

Mail Service

int-mail.itnsa.id

1. Install and configure Postfix & Dovecot service to make this server act as a mail server for **itnsa.id** domain
2. Mail server must be listen on **Submission port (TCP/587)** and **IMAP (TCP/143)**
3. Secure the mail server using certificate issued by **ITNSA-CA**
 - You can use any certificate subject common name as long it is valid, even wildcard too
4. Make sure LDAP user can be used for this mail service

VPN Service

fw.itnsa.id

1. Install OpenVPN service on this server
2. Configure OpenVPN service with specification refer to the list below:
 - Save VPN server configuration as **/etc/openvpn/vpn-server.conf**
 - Use **tun** interface type
 - Listen on **UDP port 19994**
 - Use subnet **172.16.0.0/24**
3. Save OpenVPN log into **/var/log/vpn.log** file

Firewall Filter & NAT

fw.itnsa.id

1. Configure firewall for filtering and NAT using nftables and make sure it persist across reboot
2. Configure filtering rules refer to the list below
 - Set INPUT & FORWARD chain policy to **DROP**
 - Create **input** chain rule:
 - Use layer 4 for filtering service, e.g: `tcp dport X OR udp dport X`
 - Permit OpenVPN traffic UDP/19994 to 138.24.11.24/24 from anywhere
 - Permit HTTP & HTTPS traffic to 138.24.11.24/24 from anywhere
 - Permit DNS traffic to 138.24.11.24/24 from anywhere
 - Permit TCP/9000 from anywhere for marking purpose
 - Create **forward** chain rule:
 - Use layer 4 for filtering service, e.g: `tcp dport X OR udp dport X`
 - Permit Mail Traffic (Submission & IMAP) from anywhere into int-mail (192.168.1.1/24)
 - Permit DNS traffic from anywhere into 192.168.2.100/24
 - Permit HTTP & HTTPS traffic from anywhere into 192.168.2.100/24
 - Permit traffic from INT (192.168.1.0/24) and DMZ (192.168.2.0/24) and vice-versa
 - Permit traffic from VPN subnet (172.16.0.0/24) to INT & DMZ
3. Configure NAT rules refer to the list below:
 - HTTP traffic destined to 138.24.11.24 must be forwarded to 192.168.2.100
 - HTTPS traffic destined to 138.24.11.24 must be forwarded to 192.168.2.100
 - DNS traffic destined to 138.24.11.24 must be forwarded to 192.168.2.100
 - Masquerade any internal traffic (192.168.1.0/24, 192.168.2.0/24) destined to internet into public IP of fw (138.24.11.24)

Client Configuration

budi-client

1. This client IP address, DNS and gateway are already preconfigured. **DO NOT CHANGE OR MODIFY IT!**
2. Make sure Root CA and Sub CA of **itnsa.id** already installed on this host (both on browser and local system `/usr/local/share/ca-certificates`)
3. Install OpenVPN and test VPN connection of **itnsa.id**

- Create shell script **/opt/vpn/up.sh** to connect into VPN of **itnsa.id**
 - Also create **/opt/vpn/down.sh** to disconnect from VPN of **itnsa.id**
 - Make sure this script can be executed by non-root user
4. Install Thunderbird and setup the mail of **itnsa.id** domain
- Install Root CA and Sub CA of **itnsa.id**
 - Setup account of **worker01** then try to send email to **worker01** too

Ansible Automation

int-mail.itnsa.id

1. Ansible Setup

Setup Ansible on this server with specification refer to the list below:

- Create directory **/opt/ansible** as the root directory of Ansible
 - Any playbook will be executed inside this directory, so make sure the directory name and ansible configuration is valid
- Create inventory then add **web01.itnsa.id** and **web02.itnsa.id** server. You can use any authentication method to connect the server
- After that, test the ansible connection using command below:

```
ansible web01 -m ping
ansible web02 -m ping
```

2. Initial Playbook

Create initial playbook as the initial step of ansible:

- Name the playbook as **/opt/ansible/01-initial.yml** that target both web servers
- When this playbook executed, it must install package **iftop**, **bttop** and **nethogs** then create directory **/opt/logs**

3. Virtualhost Deployment

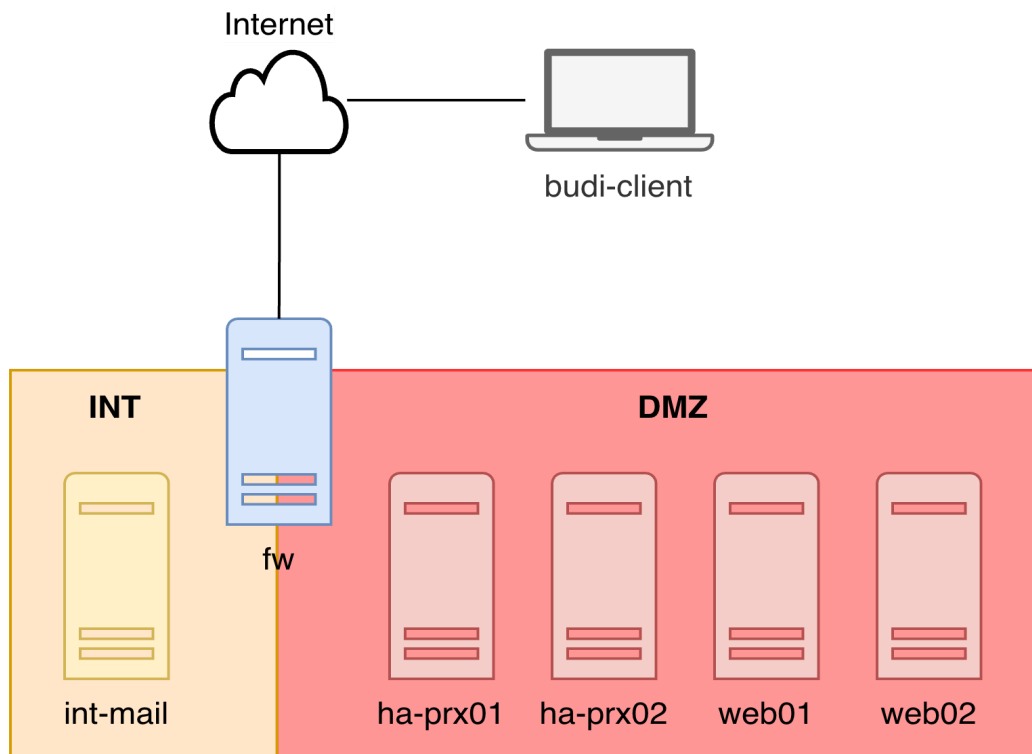
Create a playbook to automatically deploy virtualhost on both web servers.

- Name the playbook as **/opt/ansible/02-deploy-web.yml** that target both web servers
- The playbook should accept the following variable **<domain>** as the website name.
- Deployed virtualhost must use **/data/www/<domain>** as the root directory
- Set virtualhost default content into **<domain> site**
- Example execution:

```
ansible-playbook 02-deploy-web.yml -e "domain=test.itnsa.id"
```

Appendix

Topology



Addressing Table

Hostname	FQDN	IP Address
int-mail	int-mail.itnsa.id	Management: 10.10.10.1/24 ens192: 192.168.1.1/24
fw	fw.itnsa.id	Management: 10.10.10.2/24 ens192: 138.24.11.24/24 ens224: 192.168.1.254/24 ens256: 192.168.2.254/24
ha-prx01	ha-prx01.itnsa.id	Management: 10.10.10.3/24 ens192: 192.168.2.10/24
ha-prx02	ha-prx02.itnsa.id	Management: 10.10.10.4/24 ens192: 192.168.2.20/24

web01	web01.itnsa.id	Management: 10.10.10.5/24 ens192: 192.168.2.101/24
web02	web02.itnsa.id	Management: 10.10.10.6/24 ens192: 192.168.2.102/24
budi-client	budi-client	<i>*Preconfigured</i> Management: 10.10.10.7/24 ens192: 138.24.11.111/24