BUKU BAHAN AJAR GURU

ADMINISTRASI SISTEM JARINGAN

Kelas XI — Program Keahlian Teknik Jaringan Komputer dan Telekomunikasi (TJKT)

Dilengkapi Uraian Materi, Praktik Konfigurasi Server, dan Contoh Kehidupan Nyata
Disusun berdasarkan Naskah Soal LKS SMK Tingkat Kabupaten Tasikmalaya Tahun 2026
Bidang IT Network System Administration — Modul A: Linux Environment

Studi Kasus: Infrastruktur Perusahaan ITNSA.ID

Yusuf Burhani, S.Pd., S.Kom

Kata Pengantar

Buku ini disusun sebagai bahan ajar bagi guru mata pelajaran Administrasi Sistem Jaringan pada Program Keahlian Teknik Jaringan Komputer dan Telekomunikasi (TJKT) Kelas XI di Sekolah Menengah Kejuruan. Penyusunan materi mengacu pada dokumen Target Pembelajaran Administrasi Sistem Jaringan yang disusun berdasarkan naskah soal Lomba Kompetensi Siswa (LKS) SMK Tingkat Kabupaten Tasikmalaya Tahun 2026, Bidang IT Network System Administration, Modul A: Linux Environment.

Naskah soal LKS tersebut menyajikan studi kasus infrastruktur perusahaan fiktif bernama ITNSA.ID, lengkap dengan server internal (int-srv), server DMZ (mail, web-01, web-02), firewall/gateway, dan client. Studi kasus inilah yang dijadikan benang merah (running case) di sepanjang buku ini, sehingga peserta didik tidak mempelajari layanan server secara terpisah-pisah, melainkan sebagai bagian dari satu infrastruktur perusahaan yang saling terintegrasi — persis seperti yang akan mereka hadapi di dunia kerja maupun kompetisi keahlian.

Mata pelajaran Administrasi Sistem Jaringan berfokus pada kemampuan memasang, mengonfigurasi, dan mengelola berbagai layanan server agar dapat beroperasi dan diakses sesuai fungsinya (service delivery). Hal ini perlu ditegaskan sejak awal agar tidak tumpang tindih dengan mata pelajaran Keamanan Jaringan yang menitikberatkan pada pengerasan (hardening), deteksi ancaman, dan mitigasi risiko terhadap layanan yang sudah berjalan. Dengan kata lain, pada mata pelajaran ini peserta didik belajar MEMBANGUN agar layanan berfungsi; pada mata pelajaran Keamanan Jaringan peserta didik belajar MENGAMANKAN layanan yang telah berjalan tersebut.

Buku ini terdiri atas dua belas bab yang disusun mengikuti alur infrastruktur pada soal LKS, dimulai dari fondasi internal (DNS, LDAP, dan PKI/CA), dilanjutkan Mail Server, Web Server dan High Availability tingkat aplikasi, Otomasi menggunakan Ansible, Gateway/Firewall dan VPN, hingga pengujian dari sisi klien. Dua bab tambahan (Bab 10 dan Bab 11) memperkenalkan teknologi virtualisasi Proxmox Virtual Environment (VE) sebagai platform yang dapat menjalankan seluruh layanan ITNSA.ID, sekaligus konsep High Availability pada tingkat infrastruktur melalui cluster Proxmox — materi ini bersumber dari Panduan Praktikum Administrasi Sistem Jaringan – Proxmox VE 8.0 karya I Putu Hariyadi serta dokumentasi resmi Proxmox VE. Bab terakhir menyajikan proyek terintegrasi yang menggabungkan seluruh layanan menjadi satu simulasi topologi utuh, sebagaimana skenario lengkap pada soal LKS.

Setiap bab disusun dengan struktur konsisten: tujuan pembelajaran, uraian materi konseptual, kotak praktik berisi perintah konfigurasi nyata pada Debian Linux, kotak contoh kehidupan nyata untuk menjembatani teori dengan praktik dunia kerja, aktivitas pembelajaran, rangkuman, serta soal latihan/refleksi. Semoga buku ini bermanfaat sebagai pegangan guru dalam mempersiapkan peserta didik menjadi administrator jaringan yang kompeten dan siap menghadapi kompetisi keahlian maupun dunia kerja.

Penyusun

Daftar Isi

Peta Kompetensi Buku

Tabel berikut memetakan setiap elemen kompetensi pada dokumen Target Pembelajaran Administrasi Sistem Jaringan terhadap jumlah target kompetensi dan fokus utamanya. Seluruh elemen berstatus wajib karena bersumber langsung dari naskah soal LKS SMK Bidang IT Network System Administration.

No	Elemen	Jumlah Target	Fokus Utama
1	Administrasi DNS Server	3	Instal BIND9, forward/reverse zone, record A/CNAME/MX
2	Administrasi Direktori Terpusat (LDAP)	2	Instal slapd, kelola akun pengguna terpusat
3	Administrasi PKI/CA	3	Root CA, penerbitan sertifikat, pengelolaan penyimpanan sertifikat
4	Administrasi Mail Server	5	Postfix, Dovecot, TLS, integrasi LDAP, pengujian kirim/terima
5	Administrasi Web Server & High Availability	4	Nginx, Keepalived (VRRP), HAProxy (load balancing & TLS termination)
6	Otomasi Administrasi Server	3	Ansible control node, playbook user, playbook deployment
7	Administrasi Gateway (Firewall Fungsional & VPN)	4	NAT masquerade, port forwarding, WireGuard VPN, routing & DNS client
8	Administrasi dan Pengujian Sisi Klien	4	User lokal, koneksi VPN, uji web/SSL, uji mail client
9	Virtualisasi Server (Proxmox VE)	4	Instalasi Proxmox, KVM & LXC, user/permission, backup/restore, firewall & Fail2ban
10	High Availability Virtualization (Proxmox Cluster)	4	Cluster, quorum, fencing/watchdog, HA Manager, HA Group, live migration

BAB 1 — Pendahuluan dan Studi Kasus Infrastruktur ITNSA.ID

Tujuan Pembelajaran

- Guru dan peserta didik memahami ruang lingkup mata pelajaran Administrasi Sistem Jaringan serta perbedaannya dengan mata pelajaran Keamanan Jaringan.
- Peserta didik memahami gambaran umum studi kasus infrastruktur perusahaan ITNSA.ID yang akan digunakan sebagai benang merah di seluruh bab.
- Peserta didik memahami alur pembelajaran dan lingkungan praktik yang akan digunakan sepanjang semester.

1.1 Ruang Lingkup Mata Pelajaran

Administrasi Sistem Jaringan adalah mata pelajaran Kelas XI Program Keahlian TJKT yang berfokus pada kemampuan memasang, mengonfigurasi, dan mengelola berbagai layanan server agar dapat beroperasi dan diakses sesuai fungsinya. Berbeda dengan mata pelajaran Sistem Komputer di Kelas X yang menitikberatkan pada fondasi teori (sistem bilangan, logika digital, arsitektur komputer, perangkat keras), mata pelajaran ini sudah masuk ke ranah penerapan langsung: menginstal layanan DNS, mengelola direktori pengguna terpusat, menerbitkan sertifikat digital, mengoperasikan mail server, menjaga ketersediaan web server, mengotomasi konfigurasi, hingga membangun gateway dan VPN.

1.2 Batas dengan Mata Pelajaran Keamanan Jaringan

Agar tidak terjadi tumpang tindih materi antar mata pelajaran, pembagian fokus ditetapkan secara tegas sebagai berikut.

Mata Pelajaran	Fokus Utama
Administrasi Sistem Jaringan	Instalasi, konfigurasi fungsional, dan operasional layanan agar berjalan sesuai kebutuhan (service delivery) — MEMBANGUN agar layanan berfungsi.
Keamanan Jaringan	Pengerasan (hardening), deteksi ancaman, dan mitigasi risiko terhadap layanan yang sudah berjalan — MENGAMANKAN layanan yang telah dibangun.

Sebagai ilustrasi, pada mata pelajaran ini peserta didik belajar memasang dan mengonfigurasi Postfix agar dapat mengirim dan menerima email (Bab 5). Pada mata pelajaran Keamanan Jaringan, peserta didik baru akan mempelajari cara mendeteksi dan mencegah email spam/phishing yang masuk melalui server tersebut. Pemisahan fokus ini membuat peserta didik memahami siklus hidup layanan server secara utuh: dibangun terlebih dahulu agar berfungsi, kemudian diperkuat agar aman.

1.3 Studi Kasus: Infrastruktur Perusahaan ITNSA.ID

Seluruh bab dalam buku ini menggunakan satu studi kasus berkelanjutan (running case), yaitu infrastruktur jaringan perusahaan fiktif bernama ITNSA.ID. Pendekatan ini dipilih karena mencerminkan skenario nyata pada naskah soal LKS SMK Bidang IT Network System Administration, di mana peserta didik tidak

mengonfigurasi layanan secara terisolasi, melainkan sebagai bagian dari satu infrastruktur perusahaan yang saling terhubung dan saling bergantung.

Topologi dasar infrastruktur ITNSA.ID terdiri atas beberapa segmen jaringan dan server berikut.

Segmen/Server	Peran dalam Infrastruktur
int-srv (Server Internal)	Menjalankan layanan internal perusahaan: DNS internal, LDAP (direktori pengguna terpusat), dan Certificate Authority (CA) internal.
mail (Server DMZ)	Menjalankan layanan email perusahaan (Postfix untuk SMTP, Dovecot untuk IMAP) yang dapat diakses dari luar melalui gateway.
web-01 & web-02 (Server DMZ)	Dua server web (Nginx) yang dikonfigurasi secara High Availability menggunakan Keepalived dan HAProxy agar layanan tetap tersedia meski salah satu server mati.
Firewall/Gateway	Mengatur NAT, port forwarding dari internet menuju layanan DMZ, serta menjadi server VPN (WireGuard) bagi client eksternal.
Client	Perangkat pengguna akhir (baik di jaringan internal maupun melalui VPN dari luar) yang mengakses seluruh layanan di atas.

Sumber: Diadaptasi dari Naskah Soal LKS SMK Tingkat Kabupaten Tasikmalaya Tahun 2026, Bidang IT Network System Administration, Modul A: Linux Environment.

Perlu dicatat bahwa seluruh server/VM pada tabel di atas (Bab 2 hingga Bab 9) dapat dijalankan pada perangkat lunak virtualisasi apa pun, termasuk VMware Workstation yang digunakan sebagai lingkungan praktik utama pada bab-bab tersebut. Pada Bab 10 dan Bab 11, buku ini memperkenalkan Proxmox Virtual Environment (VE) sebagai platform virtualisasi kelas produksi yang mampu menjalankan seluruh VM/container ITNSA.ID di atas, sekaligus menambahkan lapisan High Availability agar layanan tetap tersedia meski terjadi kegagalan perangkat keras server fisik — melengkapi gambaran infrastruktur ITNSA.ID secara menyeluruh, mulai dari layanan aplikasi hingga platform virtualisasi yang menopangnya.

1.4 Lingkungan Praktik

Elemen 1 hingga 6 (DNS, LDAP, PKI/CA, Mail Server, Web Server & High Availability, dan Otomasi Ansible) sebaiknya dipraktikkan pada server internal/DMZ menggunakan VMware Workstation dengan topologi menyerupai soal LKS (int-srv, mail, web-01, web-02), agar peserta didik terbiasa dengan skenario multi-server sejak awal. Sistem operasi yang digunakan pada seluruh server adalah Debian Linux, sedangkan konfigurasi dilakukan melalui command line interface (CLI) agar peserta didik memahami setiap parameter yang diketik, bukan sekadar mengklik menu grafis.

1.5 Alur Pembelajaran yang Disarankan

Urutan pembelajaran yang disarankan mengikuti alur infrastruktur pada soal LKS: DNS & LDAP & CA (fondasi internal) → Mail Server → Web Server & High Availability → Otomasi (Ansible) → Gateway/Firewall & VPN → Pengujian sisi klien. Urutan ini penting karena beberapa layanan saling bergantung — misalnya Mail Server pada Bab 5 memerlukan DNS (Bab 2) untuk resolusi nama domain, LDAP (Bab 3) untuk autentikasi pengguna, dan sertifikat dari CA (Bab 4) untuk enkripsi TLS.

Contoh dalam Kehidupan Nyata: Mengapa urutan pembelajaran ini penting?

> *Bayangkan seorang teknisi baru diminta memasang mail server tanpa memahami DNS terlebih dahulu. Ia akan kebingungan saat diminta membuat MX record, karena MX record adalah bagian dari konfigurasi DNS yang mengarahkan email domain ke server mail yang tepat. Begitu pula, memasang mail server tanpa memahami PKI/CA akan membuat teknisi kesulitan saat diminta mengaktifkan enkripsi TLS pada email, karena TLS memerlukan sertifikat digital yang diterbitkan oleh Certificate Authority. Inilah sebabnya buku ini disusun secara berjenjang mengikuti dependensi teknis yang sesungguhnya terjadi di infrastruktur nyata.*

Aktivitas Pembelajaran

1. Peserta didik menggambar ulang topologi infrastruktur ITNSA.ID (int-srv, mail, web-01, web-02, firewall/gateway, client) beserta segmen jaringan (internal, DMZ, eksternal) dalam bentuk diagram.

2. Diskusi kelompok mengidentifikasi layanan apa saja yang akan dijalankan pada masing-masing server, serta menentukan alamat IP sementara untuk setiap segmen.

3. Instalasi awal empat mesin virtual Debian (int-srv, mail, web-01, web-02) pada VMware Workstation sebagai persiapan praktik bab-bab berikutnya.

Rangkuman

Mata pelajaran Administrasi Sistem Jaringan berfokus pada instalasi dan konfigurasi fungsional layanan server (service delivery), berbeda dengan Keamanan Jaringan yang berfokus pada hardening. Seluruh bab dalam buku ini menggunakan studi kasus berkelanjutan infrastruktur ITNSA.ID yang terdiri atas int-srv, mail, web-01, web-02, firewall/gateway, dan client. Alur pembelajaran disusun mengikuti dependensi teknis nyata: fondasi internal (DNS-LDAP-CA) terlebih dahulu, kemudian Mail Server, Web Server & HA, Otomasi, Gateway/VPN, dan diakhiri pengujian sisi klien.

Soal Latihan / Refleksi

1. Jelaskan perbedaan fokus antara mata pelajaran Administrasi Sistem Jaringan dan Keamanan Jaringan.

2. Sebutkan lima server/segmen utama dalam studi kasus infrastruktur ITNSA.ID beserta perannya masing-masing.

3. Jelaskan mengapa DNS perlu dipelajari sebelum Mail Server dalam alur pembelajaran buku ini.

BAB 2 — Administrasi DNS Server (BIND9)

Tujuan Pembelajaran

- Peserta didik mampu menginstal dan mengonfigurasi DNS Server (BIND9) pada Debian.
- Peserta didik mampu mengelola zona DNS, baik forward zone maupun reverse zone.
- Peserta didik mampu mengelola berbagai jenis record DNS (A, CNAME, MX) sesuai kebutuhan infrastruktur.

2.1 Konsep Dasar DNS

Domain Name System (DNS) adalah layanan yang menerjemahkan nama domain yang mudah diingat manusia (misalnya mail.itnsa.id) menjadi alamat IP yang dipahami komputer (misalnya 10.10.12.5), maupun sebaliknya. Tanpa DNS, pengguna harus menghafal alamat IP setiap layanan yang ingin diakses. Pada infrastruktur ITNSA.ID, server int-srv berperan sebagai DNS internal yang menyediakan resolusi nama untuk seluruh layanan di dalam perusahaan, seperti mail.itnsa.id, web.itnsa.id, dan ldap.itnsa.id.

2.2 Instalasi BIND9 pada Debian

BIND9 (Berkeley Internet Name Domain, versi 9) adalah perangkat lunak DNS server open-source yang paling banyak digunakan di lingkungan Linux, termasuk pada studi kasus LKS Bidang IT Network System Administration. Instalasi dilakukan melalui manajer paket apt pada Debian.

```
root@int-srv:~# apt update
root@int-srv:~# apt install bind9 bind9utils bind9-doc -y
root@int-srv:~# systemctl status bind9
root@int-srv:~# systemctl enable bind9
```

2.3 Struktur Konfigurasi BIND9

Konfigurasi BIND9 pada Debian tersebar di beberapa berkas utama pada direktori /etc/bind/. Berkas named.conf.options mengatur perilaku global server, named.conf.local mengatur daftar zona yang dikelola, sedangkan berkas zona (misalnya db.itnsa.id) berisi record DNS yang sesungguhnya.

Berkas	Fungsi
/etc/bind/named.conf.options	Pengaturan global: forwarders, listen address, opsi keamanan dasar
/etc/bind/named.conf.local	Daftar zona (forward dan reverse) yang dikelola server ini
/etc/bind/db.itnsa.id	Berkas forward zone: memetakan nama domain ke alamat IP
/etc/bind/db.10.10.12	Berkas reverse zone: memetakan alamat IP ke nama domain

2.4 Mengelola Forward Zone

Forward zone digunakan untuk menerjemahkan nama domain menjadi alamat IP. Langkah pertama adalah mendaftarkan zona pada named.conf.local, kemudian membuat berkas zona itu sendiri.

```
// /etc/bind/named.conf.local
zone "itnsa.id" {
    type master;
```

```
    file "/etc/bind/db.itnsa.id";
};

; /etc/bind/db.itnsa.id
$TTL    604800
@   IN  SOA  int-srv.itnsa.id. admin.itnsa.id. (
                  2         ; Serial
             604800         ; Refresh
              86400         ; Retry
            2419200         ; Expire
             604800 )       ; Negative Cache TTL
@       IN  NS      int-srv.itnsa.id.
int-srv IN  A       10.10.12.2
mail    IN  A       10.10.12.5
web     IN  A       10.10.12.10
```

2.5 Mengelola Reverse Zone

Reverse zone bekerja sebaliknya, yaitu menerjemahkan alamat IP menjadi nama domain — sering digunakan untuk verifikasi keabsahan server email (reverse DNS lookup) maupun keperluan audit jaringan.

```
// /etc/bind/named.conf.local (tambahan)
zone "12.10.10.in-addr.arpa" {
    type master;
    file "/etc/bind/db.10.10.12";
};

; /etc/bind/db.10.10.12
$TTL    604800
@   IN  SOA  int-srv.itnsa.id. admin.itnsa.id. ( 2 604800 86400 2419200 604800 )
@       IN  NS      int-srv.itnsa.id.
2       IN  PTR     int-srv.itnsa.id.
5       IN  PTR     mail.itnsa.id.
10      IN  PTR     web.itnsa.id.
```

2.6 Jenis-Jenis Record DNS

Setiap layanan pada infrastruktur ITNSA.ID memerlukan jenis record DNS yang berbeda sesuai fungsinya.

Jenis Record	Fungsi	Contoh Penggunaan pada ITNSA.ID
A	Memetakan nama domain ke alamat IPv4	mail.itnsa.id → 10.10.12.5
CNAME	Membuat alias/nama alternatif menuju nama domain lain	www.itnsa.id → web.itnsa.id
MX	Menentukan server mana yang menangani email untuk suatu domain, lengkap dengan prioritas	itnsa.id → mail.itnsa.id (priority 10), dipakai Postfix pada Bab 5
PTR	Memetakan alamat IP kembali ke nama domain (reverse)	10.10.12.5 → mail.itnsa.id
NS	Menunjuk server otoritatif yang mengelola suatu zona	itnsa.id → int-srv.itnsa.id

```
; Menambahkan CNAME dan MX pada db.itnsa.id
www     IN  CNAME   web.itnsa.id.
itnsa.id.  IN  MX  10  mail.itnsa.id.
```

2.7 Verifikasi dan Pengujian

Setelah konfigurasi diubah, wajib dilakukan pemeriksaan sintaks sebelum me-restart layanan, agar layanan DNS tidak gagal berjalan akibat kesalahan penulisan berkas zona.

```
root@int-srv:~# named-checkconf
root@int-srv:~# named-checkzone itnsa.id /etc/bind/db.itnsa.id
root@int-srv:~# systemctl restart bind9
root@int-srv:~# dig @10.10.12.2 mail.itnsa.id
root@int-srv:~# dig @10.10.12.2 -x 10.10.12.5
```

Praktik: Membangun DNS Internal untuk ITNSA.ID

1. Pada mesin virtual int-srv, instal BIND9 mengikuti langkah pada bagian 2.2.

2. Buat forward zone untuk domain itnsa.id dengan record A untuk int-srv, mail, web-01, dan web-02 sesuai alamat IP masing-masing pada lab.

3. Buat reverse zone untuk rentang IP internal (misalnya 10.10.12.0/24) dengan record PTR yang sesuai.

4. Tambahkan record CNAME www yang mengarah ke web.itnsa.id, dan record MX yang mengarah ke mail.itnsa.id dengan priority 10.

5. Jalankan named-checkzone untuk memvalidasi berkas zona, lalu restart layanan BIND9.

6. Dari client, arahkan DNS server ke alamat IP int-srv, lalu uji dengan perintah dig atau nslookup untuk memastikan seluruh record dapat di-resolve dengan benar.

Contoh dalam Kehidupan Nyata: DNS sebagai "buku telepon" internet

> *DNS dapat dianalogikan seperti buku telepon raksasa: manusia mengingat nama ("Kantor Pusat ITNSA"), sedangkan sistem telepon sesungguhnya menghubungkan berdasarkan nomor. Setiap kali seseorang mengetik www.itnsa.id di browser, di belakang layar terjadi proses seperti membuka buku telepon tersebut untuk menemukan "nomor" (alamat IP) dari web-01. Ketika perusahaan berpindah server atau mengganti IP publik, admin hanya perlu memperbarui satu entri DNS — seluruh pengguna tetap bisa mengakses dengan nama domain yang sama tanpa perlu tahu perubahan IP di baliknya. Inilah mengapa hampir semua layanan digital modern, mulai dari email korporat hingga aplikasi perbankan, sangat bergantung pada DNS yang dikonfigurasi dengan benar.*

Aktivitas Pembelajaran

1. Praktik instalasi BIND9 dan pembuatan forward zone serta reverse zone untuk domain itnsa.id.

2. Praktik menambahkan record A, CNAME, dan MX sesuai kebutuhan infrastruktur pada studi kasus.

3. Pengujian resolusi nama menggunakan dig/nslookup dari sisi client, mendokumentasikan hasilnya sebagai bukti kerja.

4. Diskusi kelompok: apa yang akan terjadi pada layanan email dan web jika record MX atau CNAME salah dikonfigurasi?

Rangkuman

DNS Server (BIND9) menjadi fondasi pertama infrastruktur ITNSA.ID karena hampir seluruh layanan lain (mail, web, LDAP) bergantung pada resolusi nama domain. Konfigurasi BIND9 melibatkan pengelolaan forward zone (nama ke IP) dan reverse zone (IP ke nama), serta berbagai jenis record seperti A, CNAME, MX, PTR, dan NS sesuai kebutuhan masing-masing layanan. Verifikasi sintaks menggunakan named-checkzone sebelum restart layanan adalah kebiasaan wajib bagi seorang administrator jaringan.

Soal Latihan / Refleksi

1. Jelaskan perbedaan fungsi forward zone dan reverse zone pada BIND9.

2. Buatlah contoh record MX untuk domain itnsa.id yang mengarah ke mail.itnsa.id dengan priority 10.

3. Jelaskan mengapa perintah named-checkzone perlu dijalankan sebelum me-restart layanan BIND9.

4. Jelaskan kegunaan record PTR beserta satu contoh skenario di mana record ini penting, misalnya dalam konteks reputasi pengiriman email.

BAB 3 — Administrasi Direktori Terpusat (LDAP)

Tujuan Pembelajaran

- Peserta didik mampu menginstal dan mengonfigurasi LDAP Server (slapd) sebagai basis manajemen akun terpusat.
- Peserta didik mampu mengelola akun pengguna pada LDAP (nama, uid, password, organizational unit, email) yang akan digunakan untuk autentikasi layanan lain.

3.1 Konsep Direktori Terpusat

Pada perusahaan dengan banyak layanan server — seperti ITNSA.ID yang memiliki mail server, web server, dan berbagai server internal lainnya — akan sangat merepotkan apabila setiap layanan memiliki basis data pengguna sendiri-sendiri. Karyawan harus mengingat banyak kombinasi username dan password berbeda, sementara admin harus membuat dan menghapus akun secara berulang di setiap server saat ada karyawan baru atau yang keluar. LDAP (Lightweight Directory Access Protocol) hadir untuk mengatasi masalah ini dengan menyediakan satu direktori pusat yang menyimpan seluruh identitas pengguna, sehingga layanan lain (misalnya mail server pada Bab 5) cukup "bertanya" ke LDAP untuk memverifikasi identitas seseorang.

3.2 Instalasi slapd pada Debian

slapd (Standalone LDAP Daemon) adalah implementasi server LDAP dari proyek OpenLDAP yang umum digunakan di lingkungan Linux. Instalasi dilakukan pada server int-srv sebagai basis autentikasi terpusat infrastruktur ITNSA.ID.

```
root@int-srv:~# apt update
root@int-srv:~# apt install slapd ldap-utils -y
root@int-srv:~# dpkg-reconfigure slapd
  Omit OpenLDAP server configuration?              -> No
  DNS domain name:                                  -> itnsa.id
  Organization name:                                -> ITNSA
  Administrator password:                           -> (isi password kuat)
  Database backend to use:                          -> MDB
  Do you want the database to be removed on purge?  -> No
```

Proses dpkg-reconfigure di atas secara otomatis membentuk struktur direktori dasar berdasarkan domain itnsa.id, yang akan diterjemahkan menjadi base Distinguished Name (DN) dc=itnsa,dc=id.

3.3 Struktur Direktori LDAP

LDAP menyimpan data secara hierarkis menyerupai struktur pohon (tree), di mana setiap entri memiliki Distinguished Name (DN) yang unik. Struktur dasar direktori ITNSA.ID dapat digambarkan sebagai berikut.

Komponen	Contoh Nilai	Keterangan
Base DN	dc=itnsa,dc=id	Titik akar (root) dari seluruh direktori
Organizational Unit (OU)	ou=people,dc=itnsa,dc=id	Wadah pengelompokan entri, misalnya kelompok pengguna/karyawan
User Entry (uid)	uid=budi,ou=people,dc=itnsa,dc=id	Entri akun pengguna individual beserta atributnya

3.4 Membuat Organizational Unit dan Akun Pengguna

Data pada LDAP umumnya ditambahkan menggunakan berkas berformat LDIF (LDAP Data Interchange Format), kemudian diimpor menggunakan perintah ldapadd. Berikut contoh membuat OU "people" beserta satu akun pengguna bernama Budi.

```
# people.ldif
dn: ou=people,dc=itnsa,dc=id
objectClass: organizationalUnit
ou: people

# budi.ldif
dn: uid=budi,ou=people,dc=itnsa,dc=id
objectClass: inetOrgPerson
objectClass: posixAccount
objectClass: shadowAccount
uid: budi
cn: Budi Santoso
sn: Santoso
mail: budi@itnsa.id
uidNumber: 10001
gidNumber: 10001
homeDirectory: /home/budi
loginShell: /bin/bash
userPassword: {SSHA}hasilHashPasswordDisiniOtomatis

root@int-srv:~# ldapadd -x -D "cn=admin,dc=itnsa,dc=id" -W -f people.ldif
root@int-srv:~# ldapadd -x -D "cn=admin,dc=itnsa,dc=id" -W -f budi.ldif
root@int-srv:~# ldapsearch -x -b "dc=itnsa,dc=id" "(uid=budi)"
```

Atribut objectClass menentukan skema data apa saja yang berlaku pada entri tersebut. Kombinasi inetOrgPerson, posixAccount, dan shadowAccount dipilih karena akun ini akan digunakan tidak hanya untuk identitas dasar (nama, email), tetapi juga untuk autentikasi login sistem POSIX seperti pada layanan mail server dan client Linux.

3.5 Mengubah Password Pengguna

Password pengguna pada LDAP sebaiknya tidak ditulis sebagai teks biasa, melainkan menggunakan mekanisme hashing bawaan yang disediakan oleh utilitas ldappasswd.

```
root@int-srv:~# ldappasswd -x -D "cn=admin,dc=itnsa,dc=id" -W -S
"uid=budi,ou=people,dc=itnsa,dc=id"
New password:
Re-enter new password:
```

🔧 Praktik: Membangun Direktori Terpusat untuk ITNSA.ID

1. Pada mesin virtual int-srv, instal slapd mengikuti langkah pada bagian 3.2, dengan domain itnsa.id sesuai studi kasus.

2. Buat berkas LDIF untuk organizational unit "people", lalu impor menggunakan ldapadd.

3. Buat minimal tiga akun pengguna berbeda (misalnya untuk staf IT, staf keuangan, dan staf pemasaran) lengkap dengan atribut uid, cn, sn, mail, uidNumber, dan homeDirectory.

4. Verifikasi seluruh akun yang telah dibuat menggunakan ldapsearch, lalu dokumentasikan hasilnya.

5. Ubah password salah satu akun menggunakan ldappasswd, lalu verifikasi login berhasil menggunakan akun tersebut (akan diuji lebih lanjut saat integrasi dengan mail server pada Bab 5).

Contoh dalam Kehidupan Nyata: LDAP di balik login akun kantor dan sekolah

Konsep LDAP sesungguhnya sangat dekat dengan pengalaman sehari-hari. Saat siswa login ke akun Google Workspace for Education sekolah menggunakan satu username dan password untuk mengakses email, Google Classroom, dan Google Drive sekaligus, di baliknya bekerja prinsip direktori terpusat serupa LDAP. Demikian pula, saat karyawan kantor login ke komputer, email, dan sistem absensi menggunakan satu akun Microsoft Active Directory (yang juga menggunakan protokol LDAP di dalamnya), mereka sesungguhnya sedang memanfaatkan direktori terpusat yang mencegah admin IT harus membuat akun berulang kali di setiap sistem — persis seperti yang akan dibangun peserta didik untuk perusahaan ITNSA.ID pada bab ini.

Aktivitas Pembelajaran

1. Praktik instalasi slapd dan konfigurasi domain dasar itnsa.id.

2. Praktik membuat organizational unit dan minimal tiga akun pengguna melalui berkas LDIF.

3. Praktik mengganti password akun menggunakan ldappasswd dan verifikasi menggunakan ldapsearch.

4. Diskusi kelompok: apa risikonya jika password pada LDAP disimpan dalam bentuk teks biasa (plaintext), dan mengapa hashing password menjadi penting?

Rangkuman

LDAP menyediakan direktori terpusat yang menyimpan identitas pengguna dalam struktur hierarkis berbasis Distinguished Name (DN). Server slapd diinstal dan dikonfigurasi dengan domain dasar sesuai perusahaan (dc=itnsa,dc=id), kemudian data organizational unit dan akun pengguna ditambahkan melalui berkas LDIF menggunakan ldapadd. Direktori LDAP inilah yang nantinya menjadi basis autentikasi terpusat bagi layanan Mail Server pada Bab 5, sehingga peserta didik tidak perlu membuat akun berulang kali di setiap server.

Soal Latihan / Refleksi

1. Jelaskan fungsi utama LDAP sebagai direktori terpusat pada infrastruktur perusahaan.

2. Jelaskan makna Distinguished Name (DN) uid=budi,ou=people,dc=itnsa,dc=id secara bertingkat dari akar hingga daun.

3. Sebutkan perintah yang digunakan untuk menambahkan data LDIF ke server LDAP dan untuk mengganti password pengguna.

4. Jelaskan mengapa objectClass posixAccount diperlukan pada akun yang akan dipakai untuk login sistem, bukan hanya inetOrgPerson.

BAB 4 — Administrasi Public Key Infrastructure (PKI/CA)

Tujuan Pembelajaran

- Peserta didik mampu membangun Certificate Authority (CA) internal menggunakan OpenSSL beserta atribut dasar (Key Usage, Basic Constraints).
- Peserta didik mampu menerbitkan dan mendistribusikan sertifikat server untuk web server dan mail server yang ditandatangani oleh Root CA.
- Peserta didik mampu mengelola penyimpanan dan dokumentasi berkas sertifikat (root, web, mail).

4.1 Konsep Dasar PKI dan Sertifikat Digital

Public Key Infrastructure (PKI) adalah kerangka kerja yang mengatur penerbitan, distribusi, dan validasi sertifikat digital untuk mengamankan komunikasi menggunakan kriptografi kunci publik. Setiap sertifikat digital berisi kunci publik suatu server beserta identitasnya, yang ditandatangani secara digital oleh pihak tepercaya bernama Certificate Authority (CA). Ketika client (misalnya browser atau aplikasi email) terhubung ke server yang menggunakan HTTPS/TLS, client akan memeriksa apakah sertifikat server tersebut ditandatangani oleh CA yang dipercaya — jika ya, koneksi dianggap aman dan identitas server dianggap sah.

Pada infrastruktur ITNSA.ID, perusahaan tidak menggunakan CA publik berbayar (seperti yang digunakan situs publik di internet), melainkan membangun Root CA internal sendiri menggunakan OpenSSL. Pendekatan ini umum digunakan untuk mengamankan layanan internal perusahaan (mail server, web server internal) tanpa biaya sertifikat komersial, dengan syarat seluruh client di dalam perusahaan mempercayai (menginstal) sertifikat Root CA tersebut.

4.2 Membangun Root CA Internal dengan OpenSSL

Langkah pertama membangun PKI internal adalah membuat kunci privat Root CA, kemudian membuat sertifikat Root CA yang ditandatangani sendiri (self-signed), lengkap dengan atribut Basic Constraints (menandakan bahwa sertifikat ini adalah CA yang boleh menandatangani sertifikat lain) dan Key Usage (menentukan sertifikat ini hanya boleh dipakai untuk menandatangani sertifikat dan CRL).

```
root@int-srv:~# mkdir -p /etc/ssl/itnsa-ca/{private,certs,newcerts}
root@int-srv:~# cd /etc/ssl/itnsa-ca
root@int-srv:/etc/ssl/itnsa-ca# openssl genrsa -aes256 -out private/ca.key.pem 4096
root@int-srv:/etc/ssl/itnsa-ca# openssl req -config openssl.cnf -key private/ca.key.pem \
    -new -x509 -days 3650 -sha256 -extensions v3_ca \
    -out certs/ca.cert.pem \
    -subj "/C=ID/O=ITNSA/CN=ITNSA Root CA"
```

Bagian penting dalam berkas openssl.cnf pada seksi v3_ca adalah atribut berikut, yang menegaskan bahwa sertifikat ini adalah otoritas CA dan hanya boleh dipakai untuk menandatangani sertifikat/CRL, bukan untuk enkripsi komunikasi biasa.

```
[ v3_ca ]
basicConstraints = critical, CA:true
keyUsage = critical, digitalSignature, cRLSign, keyCertSign
subjectKeyIdentifier = hash
```

4.3 Menerbitkan Sertifikat untuk Web Server dan Mail Server

Setelah Root CA siap, langkah berikutnya adalah membuat kunci privat dan Certificate Signing Request (CSR) untuk setiap server yang memerlukan sertifikat, kemudian meminta Root CA menandatangani CSR tersebut sehingga menjadi sertifikat yang sah.

```
# Membuat kunci dan CSR untuk web server
root@web-01:~# openssl genrsa -out web.itnsa.id.key.pem 2048
root@web-01:~# openssl req -new -key web.itnsa.id.key.pem \
    -out web.itnsa.id.csr.pem \
    -subj "/C=ID/O=ITNSA/CN=web.itnsa.id"

# Di sisi CA (int-srv): menandatangani CSR menjadi sertifikat sah
root@int-srv:/etc/ssl/itnsa-ca# openssl ca -config openssl.cnf -extensions server_cert \
    -days 825 -notext -md sha256 \
    -in web.itnsa.id.csr.pem -out web.itnsa.id.cert.pem
```

Langkah yang sama diulangi untuk mail.itnsa.id agar layanan Postfix dan Dovecot pada Bab 5 dapat mengaktifkan enkripsi TLS menggunakan sertifikat resmi dari Root CA internal ITNSA.

4.4 Mengelola Penyimpanan dan Distribusi Sertifikat

Seluruh berkas sertifikat wajib disimpan dan didokumentasikan secara rapi agar mudah ditelusuri dan diperbarui saat masa berlakunya (expiry) mendekati batas. Struktur penyimpanan yang disarankan pada server CA (int-srv) adalah sebagai berikut.

Direktori/Berkas	Isi
/etc/ssl/itnsa-ca/private/	Kunci privat Root CA — wajib dijaga kerahasiaannya, tidak boleh didistribusikan
/etc/ssl/itnsa-ca/certs/ca.cert.pem	Sertifikat publik Root CA — didistribusikan ke seluruh server dan client agar dipercaya
/etc/ssl/itnsa-ca/newcerts/	Arsip seluruh sertifikat yang pernah diterbitkan CA, untuk keperluan audit
web.itnsa.id.cert.pem, mail.itnsa.id.cert.pem	Sertifikat masing-masing server, disalin ke server tujuan (web-01/web-02, mail)

Distribusi sertifikat Root CA (ca.cert.pem) ke seluruh client dilakukan agar browser/aplikasi tidak menampilkan peringatan "sertifikat tidak tepercaya" saat mengakses layanan internal ITNSA.ID — proses ini akan dipraktikkan lebih lanjut pada Bab 9 saat pengujian sisi klien.

🔧 Praktik: Membangun PKI Internal untuk ITNSA.ID

1. Pada server int-srv, buat struktur direktori Root CA dan bangkitkan kunci privat CA menggunakan openssl genrsa.

2. Buat sertifikat Root CA self-signed dengan Basic Constraints CA:true dan Key Usage keyCertSign, cRLSign.

3. Pada web-01, buat kunci privat dan CSR untuk domain web.itnsa.id, lalu kirim CSR tersebut ke int-srv untuk ditandatangani.

4. Ulangi proses yang sama untuk mail.itnsa.id.

5. Susun struktur penyimpanan sertifikat sesuai tabel pada bagian 4.4, lalu buat dokumentasi singkat (README) berisi daftar sertifikat yang telah diterbitkan beserta tanggal terbit dan tanggal kedaluwarsanya.

6. Verifikasi sertifikat yang telah diterbitkan menggunakan perintah: openssl x509 -in web.itnsa.id.cert.pem -text -noout

Contoh dalam Kehidupan Nyata: Ikon gembok di browser sebagai bukti kerja PKI

> *Setiap kali membuka situs perbankan atau e-commerce dan melihat ikon gembok kecil di sebelah alamat URL, sesungguhnya browser sedang memverifikasi rantai kepercayaan PKI: sertifikat situs tersebut ditandatangani oleh CA publik yang telah dipercaya secara global (seperti Let's Encrypt atau DigiCert), sama seperti Root CA internal ITNSA yang dibangun pada bab ini, hanya saja skalanya global. Jika sertifikat kedaluwarsa atau ditandatangani oleh CA yang tidak dikenal, browser akan menampilkan peringatan "Koneksi Anda tidak privat" — persis seperti yang akan dialami client ITNSA.ID apabila mereka belum menginstal sertifikat Root CA internal perusahaan. Inilah alasan setiap laptop kantor baru biasanya sudah terpasang sertifikat CA internal perusahaan sebelum digunakan karyawan.*

Aktivitas Pembelajaran

1. Praktik membangun Root CA internal lengkap dengan atribut Basic Constraints dan Key Usage yang benar.

2. Praktik menerbitkan sertifikat untuk web.itnsa.id dan mail.itnsa.id yang ditandatangani Root CA.

3. Praktik menyusun struktur penyimpanan dan dokumentasi sertifikat.

4. Diskusi kelompok: apa risikonya jika kunci privat Root CA (ca.key.pem) bocor ke pihak luar?

Rangkuman

PKI/CA internal dibangun menggunakan OpenSSL untuk menerbitkan sertifikat digital yang mengamankan komunikasi layanan internal ITNSA.ID tanpa memerlukan sertifikat komersial. Root CA dibuat dengan atribut Basic Constraints (CA:true) dan Key Usage (keyCertSign, cRLSign) yang tepat, kemudian digunakan untuk menandatangani Certificate Signing Request (CSR) dari web server dan mail server. Pengelolaan penyimpanan sertifikat yang rapi — memisahkan kunci privat yang rahasia dari sertifikat publik yang didistribusikan — menjadi kebiasaan wajib administrator PKI, dan menjadi prasyarat langsung bagi konfigurasi TLS pada Mail Server (Bab 5) dan Web Server (Bab 6).

Soal Latihan / Refleksi

1. Jelaskan fungsi atribut Basic Constraints CA:true pada sertifikat Root CA.

2. Jelaskan urutan langkah menerbitkan sertifikat untuk sebuah server, mulai dari pembuatan kunci privat hingga sertifikat ditandatangani CA.

3. Mengapa kunci privat Root CA (ca.key.pem) tidak boleh didistribusikan ke server atau client lain?

4. Jelaskan mengapa client perlu menginstal sertifikat Root CA internal ITNSA agar tidak muncul peringatan keamanan di browser.

BAB 5 — Administrasi Mail Server (Postfix & Dovecot)

Tujuan Pembelajaran

- Peserta didik mampu menginstal dan mengonfigurasi Postfix (SMTP) agar dapat mengirim dan menerima email untuk domain tertentu.
- Peserta didik mampu menginstal dan mengonfigurasi Dovecot (IMAP) agar pengguna dapat membaca email.
- Peserta didik mampu mengaktifkan enkripsi TLS pada layanan email menggunakan sertifikat dari Bab 4.
- Peserta didik mampu mengintegrasikan autentikasi mail server dengan LDAP dari Bab 3.
- Peserta didik mampu menguji fungsi pengiriman dan penerimaan email menggunakan utilitas command line (mailutils).

5.1 Arsitektur Mail Server: SMTP dan IMAP

Layanan email pada dasarnya terdiri atas dua protokol utama dengan tugas berbeda. SMTP (Simple Mail Transfer Protocol) bertanggung jawab mengirim email dari satu server ke server lain, sedangkan IMAP (Internet Message Access Protocol) bertanggung jawab menyediakan akses bagi pengguna untuk membaca email yang tersimpan di server. Pada infrastruktur ITNSA.ID, kedua fungsi ini dijalankan oleh dua perangkat lunak terpisah pada server mail: Postfix sebagai Mail Transfer Agent (MTA) yang menjalankan SMTP, dan Dovecot sebagai IMAP/POP3 server yang menjalankan pengambilan email oleh pengguna.

Sebelum melangkah lebih jauh, mail server ITNSA.ID sudah memiliki tiga prasyarat dari bab-bab sebelumnya: record MX pada DNS (Bab 2) yang mengarahkan domain itnsa.id ke mail.itnsa.id, direktori pengguna terpusat pada LDAP (Bab 3), dan sertifikat digital dari Root CA internal (Bab 4).

5.2 Instalasi dan Konfigurasi Dasar Postfix

```
root@mail:~# apt update
root@mail:~# apt install postfix -y
  General mail configuration type:  -> Internet Site
  System mail name:                 -> itnsa.id
```

Konfigurasi utama Postfix terletak pada berkas /etc/postfix/main.cf. Beberapa parameter penting yang perlu disesuaikan agar Postfix menerima email untuk domain itnsa.id ditunjukkan berikut.

```
# /etc/postfix/main.cf
myhostname = mail.itnsa.id
mydomain = itnsa.id
myorigin = $mydomain
mydestination = $myhostname, itnsa.id, localhost.$mydomain, localhost
mynetworks = 127.0.0.0/8, 10.10.12.0/24
home_mailbox = Maildir/

root@mail:~# systemctl restart postfix
root@mail:~# systemctl enable postfix
root@mail:~# postfix check
```

5.3 Instalasi dan Konfigurasi Dovecot (IMAP)

```
root@mail:~# apt install dovecot-core dovecot-imapd -y
```

Konfigurasi utama Dovecot tersebar pada beberapa berkas di direktori /etc/dovecot/conf.d/. Parameter penting yang perlu disesuaikan meliputi lokasi mailbox dan protokol yang diaktifkan.

```
# /etc/dovecot/conf.d/10-mail.conf
mail_location = maildir:~/Maildir

# /etc/dovecot/conf.d/10-master.conf (pastikan service imap aktif)
service imap-login {
  inet_listener imap {
    port = 143
  }
  inet_listener imaps {
    port = 993
    ssl = yes
  }
}

root@mail:~# systemctl restart dovecot
root@mail:~# systemctl enable dovecot
```

5.4 Mengaktifkan Enkripsi TLS

Sertifikat mail.itnsa.id.cert.pem dan kunci privatnya yang diterbitkan pada Bab 4 disalin ke server mail, kemudian direferensikan pada konfigurasi Postfix dan Dovecot agar seluruh komunikasi email (baik pengiriman maupun pengambilan) terenkripsi dan tidak dapat disadap di jaringan.

```
# /etc/postfix/main.cf (tambahan TLS)
smtpd_tls_cert_file = /etc/ssl/itnsa/mail.itnsa.id.cert.pem
smtpd_tls_key_file = /etc/ssl/itnsa/mail.itnsa.id.key.pem
smtpd_tls_CAfile = /etc/ssl/itnsa/ca.cert.pem
smtpd_use_tls = yes
smtpd_tls_security_level = may

# /etc/dovecot/conf.d/10-ssl.conf
ssl = required
ssl_cert = </etc/ssl/itnsa/mail.itnsa.id.cert.pem
ssl_key = </etc/ssl/itnsa/mail.itnsa.id.key.pem
ssl_ca = </etc/ssl/itnsa/ca.cert.pem
```

5.5 Integrasi Autentikasi dengan LDAP

Agar pengguna dapat login ke email menggunakan akun yang sama dengan akun LDAP pada Bab 3 (bukan akun sistem Linux terpisah), Dovecot dikonfigurasi untuk memeriksa kredensial langsung ke server LDAP int-srv setiap kali ada percobaan login.

```
# /etc/dovecot/conf.d/auth-ldap.conf.ext
passdb {
  driver = ldap
  args = /etc/dovecot/dovecot-ldap.conf.ext
}
userdb {
  driver = ldap
  args = /etc/dovecot/dovecot-ldap.conf.ext
}

# /etc/dovecot/dovecot-ldap.conf.ext
uris = ldap://int-srv.itnsa.id
dn = cn=admin,dc=itnsa,dc=id
dnpass = PasswordAdminLDAP
base = ou=people,dc=itnsa,dc=id
```

```
user_filter = (uid=%n)
pass_filter = (uid=%n)
```

Dengan konfigurasi ini, ketika pengguna "budi" mencoba login ke email menggunakan aplikasi client, Dovecot akan mengirim permintaan verifikasi ke int-srv, mencari entri uid=budi,ou=people,dc=itnsa,dc=id, dan memeriksa apakah password yang dimasukkan cocok dengan userPassword yang tersimpan di LDAP — proses ini persis menerapkan konsep direktori terpusat yang dipelajari pada Bab 3.

5.6 Pengujian dengan mailutils

Paket mailutils menyediakan utilitas command line untuk mengirim dan membaca email langsung dari terminal, sangat berguna untuk menguji fungsi mail server tanpa memerlukan aplikasi client grafis.

```
root@mail:~# apt install mailutils -y
root@mail:~# echo "Ini isi email percobaan" | mail -s "Uji Coba SMTP" budi@itnsa.id
root@mail:~# mail -f /home/budi/Maildir
  & 1
  & q
```

Pengujian dianggap berhasil apabila email yang dikirim dari satu pengguna dapat diterima dan dibaca oleh pengguna tujuan, log Postfix (/var/log/mail.log) tidak menunjukkan pesan error, dan koneksi IMAP dapat diverifikasi menggunakan openssl s_client untuk memastikan sertifikat TLS terpasang dengan benar.

```
root@mail:~# tail -f /var/log/mail.log
root@mail:~# openssl s_client -connect mail.itnsa.id:993 -CAfile /etc/ssl/itnsa/ca.cert.pem
```

Praktik: Membangun Mail Server Terintegrasi untuk ITNSA.ID

1. Pada server mail, instal dan konfigurasikan Postfix sesuai bagian 5.2, pastikan record MX pada Bab 2 sudah mengarah ke server ini.

2. Instal dan konfigurasikan Dovecot sesuai bagian 5.3, pastikan direktori Maildir setiap pengguna dapat terbentuk otomatis.

3. Salin sertifikat mail.itnsa.id dari int-srv (Bab 4) ke server mail, lalu aktifkan TLS pada Postfix dan Dovecot sesuai bagian 5.4.

4. Konfigurasikan integrasi autentikasi LDAP pada Dovecot sesuai bagian 5.5, gunakan akun yang telah dibuat pada Bab 3.

5. Kirim email percobaan antar dua akun pengguna menggunakan mailutils, verifikasi email diterima dengan benar, lalu periksa log mail server untuk memastikan tidak ada error.

6. Uji koneksi TLS menggunakan openssl s_client, pastikan sertifikat yang ditampilkan sesuai dengan yang diterbitkan Root CA ITNSA pada Bab 4.

Contoh dalam Kehidupan Nyata: Mail server perusahaan di balik email kantor

> *Setiap kali seorang karyawan mengirim email menggunakan alamat @namaperusahaan.com dari aplikasi Outlook atau Gmail versi kantor (Google Workspace), di baliknya terdapat rangkaian layanan persis seperti yang dibangun pada bab ini: DNS record MX mengarahkan email ke server mail yang tepat, mail server (seperti Postfix) menangani pengiriman melalui SMTP, IMAP server (seperti Dovecot) memungkinkan karyawan membaca email dari berbagai perangkat, TLS memastikan isi email tidak disadap saat transit, dan integrasi direktori terpusat (LDAP/Active Directory) memungkinkan karyawan menggunakan satu akun yang sama untuk login email maupun sistem lain. Ketika terjadi masalah "email tidak terkirim" di dunia kerja nyata, administrator jaringan akan menelusuri persis komponen-komponen yang telah dipelajari pada bab ini: cek DNS MX, cek log Postfix, cek autentikasi LDAP, dan cek sertifikat TLS.*

Aktivitas Pembelajaran

1. Praktik instalasi dan konfigurasi Postfix serta Dovecot secara berurutan sesuai lembar kerja.

2. Praktik mengaktifkan TLS menggunakan sertifikat dari Bab 4 dan memverifikasi menggunakan openssl s_client.

3. Praktik integrasi autentikasi Dovecot dengan LDAP dan pengujian login menggunakan akun LDAP.

4. Pengujian pengiriman dan penerimaan email menggunakan mailutils, didokumentasikan dengan tangkapan layar log.

5. Diskusi kelompok/role-play troubleshooting: guru menyuntikkan satu kesalahan konfigurasi (misalnya salah domain pada main.cf), siswa mendiagnosis penyebab email gagal terkirim melalui log.

Rangkuman

Mail server ITNSA.ID dibangun dari dua komponen utama: Postfix untuk SMTP (pengiriman email) dan Dovecot untuk IMAP (pembacaan email). Kedua layanan ini terintegrasi dengan tiga bab sebelumnya — DNS untuk record MX, PKI/CA untuk enkripsi TLS, dan LDAP untuk autentikasi terpusat — sehingga menunjukkan bagaimana layanan-layanan server dalam infrastruktur nyata saling bergantung dan saling melengkapi. Pengujian fungsi email dilakukan menggunakan mailutils dan diverifikasi melalui log server serta koneksi TLS.

Soal Latihan / Refleksi

1. Jelaskan perbedaan fungsi Postfix dan Dovecot pada arsitektur mail server.

2. Jelaskan tiga prasyarat dari bab-bab sebelumnya yang dibutuhkan agar mail server ITNSA.ID dapat berfungsi penuh.

3. Jelaskan langkah mengaktifkan TLS pada Postfix, sebutkan tiga parameter konfigurasi yang diperlukan.

4. Jelaskan bagaimana Dovecot memverifikasi login pengguna melalui LDAP, sertakan contoh filter pencarian yang digunakan.

5. Sebutkan perintah untuk mengirim email percobaan dari command line menggunakan mailutils.

BAB 6 — Administrasi Web Server dan High Availability

Tujuan Pembelajaran

- Peserta didik mampu menginstal dan mengonfigurasi web server (Nginx) dan menjalankan halaman web sederhana pada port tertentu.
- Peserta didik mampu mengonfigurasi Virtual IP dengan Keepalived.
- Peserta didik mampu mengonfigurasi load balancing dan TLS termination dengan HAProxy.
- Peserta didik mampu mengelola header HTTP kustom dan pengalihan HTTP ke HTTPS.

6.1 Mengapa Perlu High Availability?

Pada infrastruktur ITNSA.ID, layanan web dijalankan oleh dua server identik, web-01 dan web-02, bukan hanya satu. Pendekatan ini disebut High Availability (HA) — sebuah desain yang memastikan layanan tetap dapat diakses pengguna meskipun salah satu server mengalami gangguan atau sedang dalam perawatan (maintenance). Tanpa HA, satu server yang mati berarti seluruh layanan web perusahaan ikut lumpuh. Bab ini menggabungkan tiga komponen agar HA dapat bekerja: Nginx sebagai web server itu sendiri, Keepalived untuk menyediakan satu Virtual IP yang "berpindah" otomatis antar server, dan HAProxy untuk membagi trafik (load balancing) sekaligus menangani enkripsi HTTPS.

6.2 Instalasi dan Konfigurasi Nginx

```
root@web-01:~# apt update && apt install nginx -y
root@web-01:~# systemctl enable nginx

# /etc/nginx/sites-available/itnsa
server {
    listen 8080;
    server_name web.itnsa.id;
    root /var/www/itnsa;
    index index.html;
    location / {
        try_files $uri $uri/ =404;
    }
}

root@web-01:~# ln -s /etc/nginx/sites-available/itnsa /etc/nginx/sites-enabled/
root@web-01:~# echo "<h1>ITNSA.ID - web-01</h1>" > /var/www/itnsa/index.html
root@web-01:~# nginx -t
root@web-01:~# systemctl restart nginx
```

Konfigurasi yang sama diulangi pada web-02, dengan perbedaan isi halaman index.html menjadi "web-02" agar saat pengujian load balancing, peserta didik dapat membedakan server mana yang sedang melayani permintaan.

6.3 Virtual IP dengan Keepalived (VRRP)

Keepalived mengimplementasikan protokol VRRP (Virtual Router Redundancy Protocol) yang memungkinkan dua atau lebih server berbagi satu alamat IP virtual (Virtual IP/VIP). Salah satu server berperan sebagai MASTER yang memegang VIP secara aktif, sedangkan server lain berperan sebagai BACKUP yang siap mengambil alih VIP secara otomatis apabila MASTER terdeteksi mati.

```
root@web-01:~# apt install keepalived -y

# /etc/keepalived/keepalived.conf (di web-01, sebagai MASTER)
vrrp_instance VI_1 {
    state MASTER
    interface eth0
    virtual_router_id 51
    priority 100
    advert_int 1
    authentication {
        auth_type PASS
        auth_pass itnsaVRRP
    }
    virtual_ipaddress {
        10.10.20.100/24
    }
}

# /etc/keepalived/keepalived.conf (di web-02, sebagai BACKUP)
vrrp_instance VI_1 {
    state BACKUP
    interface eth0
    virtual_router_id 51
    priority 90
    advert_int 1
    authentication {
        auth_type PASS
        auth_pass itnsaVRRP
    }
    virtual_ipaddress {
        10.10.20.100/24
    }
}
```

Nilai priority menentukan server mana yang akan menjadi MASTER pada kondisi normal — semakin besar nilainya, semakin diprioritaskan. Ketika web-01 (priority 100) mati, web-02 (priority 90) akan mengambil alih VIP 10.10.20.100 secara otomatis dalam hitungan detik tanpa campur tangan manual administrator.

6.4 Load Balancing dan TLS Termination dengan HAProxy

HAProxy diletakkan di depan kedua server Nginx untuk membagi trafik masuk (load balancing) sekaligus menangani proses dekripsi HTTPS (TLS termination), sehingga sertifikat TLS cukup dipasang satu kali di HAProxy, bukan di setiap server backend.

```
root@web-01:~# apt install haproxy -y

# /etc/haproxy/haproxy.cfg
frontend itnsa_https
    bind *:443 ssl crt /etc/ssl/itnsa/web.itnsa.id.pem
    mode http
    http-request redirect scheme https unless { ssl_fc }
    http-response set-header X-Served-By ITNSA-LoadBalancer
    default_backend itnsa_web_pool

backend itnsa_web_pool
    mode http
    balance roundrobin
    server web01 10.10.20.11:8080 check
    server web02 10.10.20.12:8080 check

frontend itnsa_http
    bind *:80
```

```
    mode http
    redirect scheme https code 301
```

Berkas /etc/ssl/itnsa/web.itnsa.id.pem pada konfigurasi di atas adalah gabungan sertifikat dan kunci privat web.itnsa.id yang diterbitkan pada Bab 4 (digabung menjadi satu berkas karena format yang diminta HAProxy). Baris balance roundrobin menentukan strategi load balancing, yaitu setiap permintaan baru diteruskan bergantian ke web01 dan web02. Baris http-response set-header menambahkan header HTTP kustom X-Served-By pada setiap respons, dan baris redirect scheme https memastikan seluruh trafik HTTP dialihkan otomatis ke HTTPS.

```
root@web-01:~# haproxy -c -f /etc/haproxy/haproxy.cfg
root@web-01:~# systemctl restart haproxy
root@web-01:~# systemctl restart keepalived
```

Praktik: Membangun Web Server High Availability untuk ITNSA.ID

1. Instal dan konfigurasikan Nginx pada web-01 dan web-02, masing-masing menampilkan halaman identitas server yang berbeda agar mudah dibedakan.

2. Instal dan konfigurasikan Keepalived pada kedua server dengan satu Virtual IP bersama, satu sebagai MASTER dan satu sebagai BACKUP.

3. Matikan sementara layanan Nginx/jaringan pada server MASTER, amati VIP berpindah ke server BACKUP menggunakan perintah ip addr show pada masing-masing server.

4. Instal dan konfigurasikan HAProxy sebagai load balancer dan TLS termination di depan kedua server web menggunakan sertifikat dari Bab 4.

5. Uji akses berulang kali ke alamat https://web.itnsa.id dari client, amati header X-Served-By dan konten halaman untuk membuktikan trafik bergantian antara web-01 dan web-02.

6. Uji akses http://web.itnsa.id (tanpa s) dan pastikan otomatis dialihkan ke https://.

Contoh dalam Kehidupan Nyata: High availability di balik layanan yang "tidak pernah down"

> *Ketika mengakses situs e-commerce besar saat gelaran diskon (seperti Harbolnas), jutaan pengguna mengakses secara bersamaan tanpa situs tersebut down — inilah hasil kerja gabungan load balancing dan high availability berskala besar, prinsip yang sama seperti yang dibangun pada bab ini hanya dengan jumlah server yang jauh lebih banyak. Demikian pula, layanan perbankan digital yang tetap dapat diakses meski satu pusat data (data center) mengalami gangguan listrik, karena trafik otomatis dialihkan ke pusat data cadangan — konsep failover otomatis inilah yang direpresentasikan oleh perpindahan Virtual IP dari web-01 ke web-02 saat MASTER mati pada praktik di atas.*

Aktivitas Pembelajaran

1. Praktik instalasi Nginx pada dua server dengan konten pembeda.

2. Praktik konfigurasi Keepalived dan pengujian failover Virtual IP.

3. Praktik konfigurasi HAProxy untuk load balancing dan TLS termination.

4. Pengujian pengalihan HTTP ke HTTPS dan verifikasi header kustom pada respons.

5. Diskusi kelompok: apa konsekuensinya jika HAProxy sendiri (bukan Nginx) yang mati — apakah arsitektur ini memiliki titik kegagalan tunggal (single point of failure)?

Rangkuman

High Availability pada layanan web ITNSA.ID dibangun dari tiga lapis: Nginx sebagai web server pada masing-masing backend, Keepalived yang menyediakan Virtual IP dengan mekanisme failover otomatis berbasis VRRP, dan HAProxy yang membagi trafik (load balancing) sekaligus menangani TLS termination menggunakan

sertifikat dari Bab 4. Kombinasi ketiganya memastikan layanan web perusahaan tetap dapat diakses meski salah satu server backend mengalami gangguan.

Soal Latihan / Refleksi

1. Jelaskan peran masing-masing dari Nginx, Keepalived, dan HAProxy dalam arsitektur High Availability pada bab ini.

2. Jelaskan mekanisme kerja VRRP dalam menentukan server MASTER dan BACKUP.

3. Jelaskan fungsi TLS termination pada HAProxy dan mengapa hal ini menyederhanakan pengelolaan sertifikat.

4. Jelaskan bagaimana cara memverifikasi bahwa load balancing bekerja dengan benar antara web-01 dan web-02.

BAB 7 — Otomasi Administrasi Server dengan Ansible

Tujuan Pembelajaran

- Peserta didik mampu menginstal dan mengonfigurasi Ansible sebagai control node.
- Peserta didik mampu membuat playbook manajemen pengguna secara idempoten.
- Peserta didik mampu membuat playbook deployment layanan sederhana secara otomatis dan idempoten.

7.1 Mengapa Otomasi Diperlukan?

Setelah mempelajari Bab 2 hingga Bab 6, peserta didik telah mengonfigurasi banyak layanan secara manual satu per satu pada masing-masing server: DNS di int-srv, LDAP di int-srv, mail server di mail, web server di web-01 dan web-02. Bayangkan apabila infrastruktur ITNSA.ID berkembang menjadi puluhan atau ratusan server — mengonfigurasi setiap server secara manual akan memakan waktu sangat lama dan rentan kesalahan manusia (human error). Ansible hadir sebagai alat otomasi konfigurasi (configuration management) yang memungkinkan administrator mendefinisikan konfigurasi yang diinginkan dalam berkas playbook, kemudian menerapkannya ke banyak server sekaligus secara konsisten.

Konsep penting dalam Ansible adalah idempoten, yaitu playbook dapat dijalankan berulang kali tanpa menimbulkan efek samping yang tidak diinginkan — jika suatu kondisi sudah sesuai dengan yang didefinisikan (misalnya user sudah ada), Ansible tidak akan membuat ulang atau menduplikasi, melainkan melewatinya (skip).

7.2 Instalasi dan Konfigurasi Control Node

Ansible bekerja dengan arsitektur agentless: satu mesin berperan sebagai control node (biasanya int-srv atau mesin admin terpisah) yang terhubung ke seluruh server tujuan (managed node) melalui SSH, tanpa perlu menginstal perangkat lunak tambahan di sisi managed node.

```
root@int-srv:~# apt update && apt install ansible -y
root@int-srv:~# ansible --version
```

Daftar server yang dikelola didefinisikan pada berkas inventory. Berikut contoh inventory untuk seluruh server pada infrastruktur ITNSA.ID.

```
# /etc/ansible/hosts
[internal]
int-srv ansible_host=10.10.12.2

[dmz]
mail   ansible_host=10.10.20.5
web01  ansible_host=10.10.20.11
web02  ansible_host=10.10.20.12

[itnsa:children]
internal
dmz

root@int-srv:~# ssh-copy-id root@10.10.20.11
root@int-srv:~# ssh-copy-id root@10.10.20.12
root@int-srv:~# ansible itnsa -m ping
```

Perintah ansible itnsa -m ping menguji konektivitas control node ke seluruh server dalam grup itnsa menggunakan modul ping bawaan Ansible — jika seluruh server merespons "pong", control node siap digunakan untuk menjalankan playbook.

7.3 Playbook Manajemen Pengguna

Playbook berikut membuat user baru secara idempoten pada seluruh server DMZ, menggunakan variabel input agar dapat digunakan kembali (reusable) untuk pengguna berbeda.

```
# playbook-user.yml
---
- name: Membuat user baru secara idempoten
  hosts: dmz
  become: true
  vars:
    new_username: teknisi01
  tasks:
    - name: Pastikan user {{ new_username }} ada
      ansible.builtin.user:
        name: "{{ new_username }}"
        shell: /bin/bash
        create_home: yes
        state: present

root@int-srv:~# ansible-playbook playbook-user.yml
root@int-srv:~# ansible-playbook playbook-user.yml --extra-vars "new_username=teknisi02"
```

Karena modul ansible.builtin.user bersifat idempoten, menjalankan playbook yang sama dua kali dengan variabel yang sama tidak akan menghasilkan error atau duplikasi — Ansible akan melaporkan status "ok" (tidak ada perubahan) pada eksekusi kedua, bukan "changed".

7.4 Playbook Deployment Layanan Sederhana

Playbook berikut mengotomasi instalasi Nginx beserta konfigurasi virtual host sederhana pada server web-01 dan web-02 sekaligus, menggantikan langkah manual yang telah dilakukan pada Bab 6.

```
# playbook-webserver.yml
---
- name: Deploy Nginx dan virtual host ITNSA
  hosts: web01:web02
  become: true
  tasks:
    - name: Instal Nginx
      ansible.builtin.apt:
        name: nginx
        state: present
        update_cache: yes

    - name: Salin konfigurasi virtual host
      ansible.builtin.template:
        src: templates/itnsa.conf.j2
        dest: /etc/nginx/sites-available/itnsa

    - name: Aktifkan virtual host
      ansible.builtin.file:
        src: /etc/nginx/sites-available/itnsa
        dest: /etc/nginx/sites-enabled/itnsa
        state: link

    - name: Restart Nginx
```

```
      ansible.builtin.service:
        name: nginx
        state: restarted
        enabled: yes

root@int-srv:~# ansible-playbook playbook-webserver.yml
PLAY [Deploy Nginx dan virtual host ITNSA] ***************************
TASK [Instal Nginx] **************************************************
changed: [web01]
changed: [web02]
...
PLAY RECAP ***********************************************************
web01  : ok=4  changed=4  unreachable=0  failed=0
web02  : ok=4  changed=4  unreachable=0  failed=0
```

Praktik: Mengotomasi Konfigurasi Server ITNSA.ID dengan Ansible

1. Instal Ansible pada int-srv sebagai control node, siapkan berkas inventory berisi seluruh server ITNSA.ID (int-srv, mail, web01, web02).

2. Salin kunci SSH publik dari int-srv ke seluruh managed node menggunakan ssh-copy-id, lalu verifikasi konektivitas dengan ansible itnsa -m ping.

3. Buat playbook untuk menambahkan satu user baru secara idempoten pada seluruh server DMZ, jalankan dua kali berturut-turut, amati perbedaan status "changed" pada eksekusi pertama dan status "ok" pada eksekusi kedua.

4. Buat playbook deployment Nginx sederhana pada web01 dan web02, jalankan, lalu verifikasi kedua server dapat diakses melalui browser.

5. Sebagai tantangan tambahan, modifikasi playbook deployment agar juga menyalin sertifikat TLS dari Bab 4 ke lokasi yang sesuai di setiap server.

Contoh dalam Kehidupan Nyata: Otomasi sebagai kunci efisiensi tim IT modern

> *Tim IT perusahaan skala menengah hingga besar jarang mengelola server secara manual satu per satu, karena tidak efisien dan rawan human error saat jumlah server mencapai puluhan hingga ribuan. Perusahaan teknologi besar menggunakan alat otomasi seperti Ansible (atau alat sejenis seperti Puppet, Chef, atau Terraform untuk infrastruktur cloud) untuk memastikan seluruh server memiliki konfigurasi yang konsisten, memudahkan proses onboarding karyawan baru (misalnya otomatis membuat akun di puluhan sistem sekaligus), serta mempercepat pemulihan bencana (disaster recovery) — jika satu server rusak total, playbook yang sama dapat dijalankan pada server pengganti untuk memulihkan konfigurasi dalam hitungan menit, dibandingkan berjam-jam jika dilakukan manual.*

Aktivitas Pembelajaran

1. Praktik instalasi Ansible dan konfigurasi inventory serta autentikasi SSH ke seluruh managed node.

2. Praktik membuat dan menjalankan playbook manajemen pengguna, amati sifat idempoten dengan menjalankannya dua kali.

3. Praktik membuat dan menjalankan playbook deployment web server sederhana.

4. Diskusi kelompok: apa keuntungan dan risiko dari otomasi konfigurasi server dibandingkan konfigurasi manual?

Rangkuman

Ansible memungkinkan administrator jaringan mengotomasi konfigurasi banyak server sekaligus melalui berkas playbook berformat YAML, bekerja dengan arsitektur agentless melalui SSH dari satu control node. Konsep idempoten memastikan playbook dapat dijalankan berulang kali tanpa efek samping, sebagaimana dibuktikan pada playbook manajemen pengguna dan deployment web server yang dipraktikkan pada bab ini.

Otomasi ini menjadi keterampilan penting sebelum peserta didik melanjutkan ke konfigurasi gateway dan VPN pada bab berikutnya, di mana konsistensi konfigurasi antar perangkat menjadi semakin krusial.

Soal Latihan / Refleksi

1. Jelaskan apa yang dimaksud dengan sifat idempoten pada playbook Ansible, beserta satu contoh penerapannya.

2. Jelaskan perbedaan arsitektur agentless Ansible dengan alat otomasi yang memerlukan agent terinstal di setiap managed node.

3. Sebutkan langkah menyiapkan control node Ansible mulai dari instalasi hingga verifikasi konektivitas ke managed node.

4. Jelaskan struktur dasar sebuah playbook Ansible (hosts, become, tasks) berdasarkan contoh pada bab ini.

BAB 8 — Administrasi Gateway: Firewall Fungsional dan VPN

Tujuan Pembelajaran

- Peserta didik mampu mengonfigurasi akses internet untuk jaringan internal dan DMZ menggunakan NAT masquerade.
- Peserta didik mampu mengonfigurasi port forwarding menuju layanan DMZ.
- Peserta didik mampu memasang dan mengonfigurasi layanan VPN (WireGuard).
- Peserta didik mampu mengelola konfigurasi routing VPN dan DNS client.

8.1 Peran Gateway pada Infrastruktur ITNSA.ID

Server gateway/firewall pada infrastruktur ITNSA.ID memiliki tiga tugas fungsional utama yang akan dipelajari pada bab ini: pertama, menjadi pintu keluar (NAT) agar seluruh perangkat pada jaringan internal dan DMZ dapat mengakses internet meski hanya gateway yang memiliki IP publik; kedua, meneruskan (forward) trafik masuk dari internet menuju layanan tertentu di DMZ (misalnya web dan mail) agar dapat diakses dari luar; dan ketiga, menjadi server VPN agar karyawan yang bekerja dari luar kantor tetap dapat mengakses layanan internal secara aman. Perlu ditekankan kembali bahwa bab ini menitikberatkan pada aspek fungsional (agar ketiga hal tersebut BERFUNGSI), sedangkan pengerasan keamanan firewall secara mendalam dipelajari pada mata pelajaran Keamanan Jaringan.

8.2 NAT Masquerade untuk Akses Internet

NAT (Network Address Translation) tipe masquerade memungkinkan banyak perangkat dengan IP private (Bab 3 pada mata pelajaran Sistem Komputer Kelas X) mengakses internet menggunakan satu IP publik gateway. Konfigurasi berikut menggunakan nftables, framework firewall modern pada Debian yang menggantikan iptables lama.

```
root@gateway:~# apt install nftables -y
root@gateway:~# nft add table ip nat
root@gateway:~# nft add chain ip nat postrouting { type nat hook postrouting priority 100
\; }
root@gateway:~# nft add rule ip nat postrouting oifname "eth0" masquerade
root@gateway:~# echo 'net.ipv4.ip_forward=1' >> /etc/sysctl.conf
root@gateway:~# sysctl -p
```

Baris masquerade menerjemahkan seluruh trafik keluar yang melewati interface eth0 (interface yang terhubung ke internet) agar tampak berasal dari IP publik gateway, bukan dari IP private asli perangkat pengirim. Baris ip_forward=1 mengaktifkan kemampuan Linux untuk meneruskan paket antar interface, wajib diaktifkan agar gateway dapat berfungsi sebagai router.

8.3 Port Forwarding Menuju Layanan DMZ

Agar layanan web dan mail pada server DMZ dapat diakses dari internet, gateway perlu meneruskan trafik masuk pada port tertentu menuju IP internal server yang bersangkutan (DNAT/Destination NAT).

```
root@gateway:~# nft add table ip nat
root@gateway:~# nft add chain ip nat prerouting { type nat hook prerouting priority -100
\; }
root@gateway:~# nft add rule ip nat prerouting tcp dport 443 dnat to 10.10.20.100
```

```
root@gateway:~# nft add rule ip nat prerouting tcp dport 25 dnat to 10.10.20.5
root@gateway:~# nft add rule ip nat prerouting tcp dport 993 dnat to 10.10.20.5
```

Aturan pertama meneruskan trafik HTTPS (port 443) menuju Virtual IP HAProxy (10.10.20.100) yang telah dikonfigurasi pada Bab 6. Aturan kedua dan ketiga meneruskan trafik SMTP (port 25) dan IMAPS (port 993) menuju server mail (10.10.20.5) yang telah dikonfigurasi pada Bab 5 — inilah contoh nyata bagaimana bab-bab sebelumnya saling terhubung menjadi satu infrastruktur fungsional.

8.4 Layanan VPN dengan WireGuard

WireGuard adalah perangkat lunak VPN modern yang ringan, cepat, dan mudah dikonfigurasi dibandingkan solusi VPN lama seperti OpenVPN. WireGuard bekerja berdasarkan pasangan kunci publik-privat (mirip konsep PKI pada Bab 4, meski mekanismenya berbeda), di mana setiap peer (baik server maupun client) memiliki pasangan kuncinya sendiri.

```
root@gateway:~# apt install wireguard -y
root@gateway:~# wg genkey | tee privatekey | wg pubkey > publickey
root@gateway:~# cat privatekey
root@gateway:~# cat publickey

# /etc/wireguard/wg0.conf (di gateway)
[Interface]
Address = 10.99.0.1/24
ListenPort = 51820
PrivateKey = <privatekey-gateway>

[Peer]
# Client: laptop karyawan (dikonfigurasi pada Bab 9)
PublicKey = <publickey-client>
AllowedIPs = 10.99.0.2/32

root@gateway:~# wg-quick up wg0
root@gateway:~# systemctl enable wg-quick@wg0
root@gateway:~# nft add rule ip nat postrouting oifname "eth0" ip saddr 10.99.0.0/24
masquerade
```

Baris terakhir menambahkan aturan NAT masquerade khusus untuk trafik yang berasal dari jaringan VPN (10.99.0.0/24), agar client VPN juga dapat mengakses internet melalui gateway yang sama, selain mengakses jaringan internal perusahaan.

8.5 Routing VPN dan DNS Client

Agar client VPN dapat mengakses seluruh jaringan internal ITNSA.ID (bukan hanya gateway itu sendiri) dan menggunakan DNS internal (int-srv dari Bab 2) alih-alih DNS publik, konfigurasi AllowedIPs dan DNS pada sisi client perlu diperluas — bagian ini akan dilanjutkan konfigurasi lengkapnya pada Bab 9 dari sisi client.

```
# Tambahan pada [Peer] di gateway agar trafik ke seluruh jaringan internal diteruskan
AllowedIPs = 10.99.0.2/32

# Routing yang perlu dipastikan di gateway agar VPN dapat mencapai jaringan internal & DMZ
root@gateway:~# nft add rule ip filter forward ip saddr 10.99.0.0/24 accept
```

🔧 Praktik: Membangun Gateway Fungsional untuk ITNSA.ID

1. Pada server gateway, konfigurasikan NAT masquerade menggunakan nftables agar jaringan internal dan DMZ dapat mengakses internet, lalu uji dengan ping dari client internal ke 8.8.8.8.

2. Konfigurasikan port forwarding untuk port 443 menuju Virtual IP HAProxy (Bab 6) dan port 25/993 menuju server mail (Bab 5).

3. Instal WireGuard, bangkitkan pasangan kunci publik-privat untuk gateway, dan siapkan konfigurasi wg0.conf sesuai bagian 8.4.

4. Aktifkan interface WireGuard menggunakan wg-quick up wg0, verifikasi interface muncul menggunakan perintah wg show.

5. Tambahkan aturan NAT dan forward yang diperlukan agar client VPN dapat mengakses internet maupun jaringan internal ITNSA.ID.

6. Dokumentasikan seluruh aturan nftables yang telah dibuat menggunakan perintah nft list ruleset > firewall-itnsa.txt sebagai bukti kerja.

Contoh dalam Kehidupan Nyata: VPN kantor saat bekerja dari rumah (WFH)

> *Ketika karyawan bekerja dari rumah (Work From Home) namun tetap perlu mengakses sistem internal kantor seperti aplikasi absensi, file server, atau sistem keuangan yang tidak terhubung ke internet publik, mereka biasanya membuka aplikasi VPN kantor terlebih dahulu sebelum dapat mengaksesnya. Begitu terhubung, laptop mereka seolah-olah "berpindah" secara virtual ke dalam jaringan kantor, meski secara fisik berada jauh di rumah — inilah yang dibangun pada bab ini menggunakan WireGuard. Demikian pula, saat gateway kantor meneruskan trafik dari internet menuju server email dan website perusahaan (port forwarding), inilah mekanisme yang memungkinkan pelanggan dari luar dapat mengunjungi situs perusahaan atau mengirim email ke alamat @perusahaan.com meski server tersebut berada di jaringan internal yang tidak terekspos langsung ke internet.*

Aktivitas Pembelajaran

1. Praktik konfigurasi NAT masquerade dan pengujian akses internet dari client internal.

2. Praktik konfigurasi port forwarding menuju layanan web dan mail DMZ.

3. Praktik instalasi dan konfigurasi WireGuard di sisi gateway.

4. Diskusi kelompok: mengapa bab ini disebut "firewall fungsional", bukan "firewall keamanan" — apa bedanya dengan materi firewall pada mata pelajaran Keamanan Jaringan?

Rangkuman

Server gateway ITNSA.ID menjalankan tiga fungsi utama: NAT masquerade agar jaringan internal dan DMZ dapat mengakses internet, port forwarding (DNAT) agar layanan DMZ dapat diakses dari internet, dan WireGuard sebagai server VPN agar karyawan dari luar dapat terhubung ke jaringan internal secara aman. Ketiganya dikonfigurasi menggunakan nftables dan WireGuard pada Debian, menghubungkan seluruh layanan yang telah dibangun pada bab-bab sebelumnya (DNS, mail, web HA) menjadi satu infrastruktur yang dapat diakses baik dari dalam maupun luar jaringan perusahaan.

Soal Latihan / Refleksi

1. Jelaskan perbedaan fungsi NAT masquerade (SNAT) dan port forwarding (DNAT).

2. Jelaskan mengapa net.ipv4.ip_forward perlu diaktifkan pada server gateway.

3. Jelaskan konsep pasangan kunci publik-privat pada WireGuard dan bagaimana peer saling diautentikasi.

4. Jelaskan mengapa client VPN memerlukan aturan NAT tambahan agar dapat mengakses internet melalui gateway yang sama.

BAB 9 — Administrasi dan Pengujian Sisi Klien

Tujuan Pembelajaran

- Peserta didik mampu mengelola akun pengguna lokal dengan hak sudo pada mesin client.
- Peserta didik mampu mengonfigurasi koneksi client ke layanan VPN.
- Peserta didik mampu menguji akses layanan web internal dan publik dari client.
- Peserta didik mampu mengonfigurasi dan menguji klien email.

9.1 Mengapa Pengujian Sisi Klien Penting?

Setelah delapan bab sebelumnya berfokus membangun berbagai layanan di sisi server, bab ini membalikkan sudut pandang ke sisi pengguna akhir (end user). Sebuah layanan server yang telah dikonfigurasi dengan sempurna secara teknis tetap dianggap gagal apabila pengguna akhir tidak dapat mengaksesnya dengan lancar. Bab ini melatih peserta didik memverifikasi seluruh infrastruktur ITNSA.ID dari perspektif karyawan biasa yang menggunakan laptop untuk bekerja sehari-hari — sekaligus menjadi tahap pengujian end-to-end atas seluruh pekerjaan pada bab-bab sebelumnya.

9.2 Mengelola Akun Pengguna Lokal pada Client

Setiap laptop/PC client memerlukan akun pengguna lokal dengan hak akses administratif terbatas (sudo) agar karyawan dapat melakukan tugas administratif dasar (misalnya menginstal aplikasi VPN client) tanpa memiliki akses root penuh yang berisiko.

```
root@client:~# adduser budi
root@client:~# usermod -aG sudo budi
root@client:~# su - budi
budi@client:~$ sudo whoami
  root
```

9.3 Konfigurasi Koneksi Client ke VPN

Melanjutkan konfigurasi WireGuard di sisi gateway pada Bab 8, client perlu dibangkitkan pasangan kuncinya sendiri, kemudian didaftarkan sebagai peer baru di gateway.

```
budi@client:~$ sudo apt install wireguard -y
budi@client:~$ wg genkey | tee privatekey | wg pubkey > publickey

# /etc/wireguard/wg0.conf (di client)
[Interface]
Address = 10.99.0.2/24
PrivateKey = <privatekey-client>
DNS = 10.10.12.2

[Peer]
PublicKey = <publickey-gateway>
Endpoint = gateway.itnsa.id:51820
AllowedIPs = 10.10.0.0/16, 10.99.0.0/24
PersistentKeepalive = 25
```

Baris DNS = 10.10.12.2 memastikan client menggunakan DNS internal (int-srv, Bab 2) saat terhubung VPN, sehingga nama domain internal seperti mail.itnsa.id dan web.itnsa.id dapat di-resolve dengan benar. Baris

AllowedIPs mencakup seluruh rentang jaringan internal ITNSA.ID (10.10.0.0/16) agar client dapat mengakses semua segmen, bukan hanya gateway.

```
budi@client:~$ sudo wg-quick up wg0
budi@client:~$ sudo systemctl enable wg-quick@wg0
budi@client:~$ ping mail.itnsa.id
budi@client:~$ wg show
```

Perintah systemctl enable memastikan koneksi VPN otomatis tersambung setiap kali laptop dinyalakan (saat boot), sesuai target kompetensi yang mensyaratkan verifikasi koneksi berjalan otomatis.

9.4 Menguji Akses Web Internal dan Publik

Setelah VPN aktif, client menguji akses ke layanan web internal ITNSA.ID (yang dibangun pada Bab 6) maupun situs publik, sekaligus memverifikasi sertifikat SSL dari Root CA internal (Bab 4) diterima tanpa galat oleh browser.

```
budi@client:~$ curl -v https://web.itnsa.id
budi@client:~$ curl -v https://www.google.com
```

Agar browser tidak menampilkan peringatan "sertifikat tidak tepercaya" saat mengakses https://web.itnsa.id, sertifikat Root CA (ca.cert.pem dari Bab 4) perlu diinstal terlebih dahulu ke dalam trust store sistem operasi client.

```
budi@client:~$ sudo cp ca.cert.pem /usr/local/share/ca-certificates/itnsa-root-ca.crt
budi@client:~$ sudo update-ca-certificates
```

9.5 Konfigurasi dan Pengujian Klien Email

Tahap akhir pengujian adalah memastikan pengguna dapat mengirim dan menerima email menggunakan aplikasi mail client grafis, misalnya Thunderbird, dengan akun yang telah dibuat pada LDAP (Bab 3) dan terautentikasi melalui Dovecot (Bab 5).

Parameter	Nilai untuk ITNSA.ID
Nama Pengguna	Budi Santoso
Alamat Email	budi@itnsa.id
Server Masuk (IMAP)	mail.itnsa.id, port 993, SSL/TLS
Server Keluar (SMTP)	mail.itnsa.id, port 587/25, STARTTLS
Autentikasi	Password biasa (diverifikasi ke LDAP oleh Dovecot)

Pengujian dinyatakan berhasil apabila Thunderbird dapat login menggunakan akun LDAP (budi/password yang telah diset pada Bab 3), menerima email uji coba yang dikirim melalui mailutils pada Bab 5, serta dapat mengirim balasan yang diterima kembali oleh pengirim.

🔧 Praktik: Verifikasi End-to-End Infrastruktur ITNSA.ID dari Sisi Klien

1. Buat akun pengguna lokal baru pada mesin client dengan hak sudo.

2. Bangkitkan pasangan kunci WireGuard di client, daftarkan public key tersebut sebagai peer baru pada gateway (Bab 8), lalu aktifkan koneksi VPN dan pastikan aktif otomatis saat boot.

3. Instal sertifikat Root CA ITNSA ke trust store client, lalu uji akses https://web.itnsa.id dan pastikan tidak ada peringatan keamanan pada browser.

4. Uji akses ke satu situs publik (misalnya https://www.google.com) untuk memastikan VPN tidak mengganggu akses internet biasa.

5. Konfigurasikan Thunderbird (atau mail client lain) menggunakan akun LDAP yang telah dibuat pada Bab 3, kirim dan terima email uji coba dengan pengguna lain.

6. Susun laporan pengujian akhir berisi seluruh langkah di atas beserta tangkapan layar sebagai bukti bahwa infrastruktur ITNSA.ID dapat diakses dan digunakan sebagaimana mestinya oleh pengguna akhir.

Contoh dalam Kehidupan Nyata: Perspektif helpdesk IT: "Kenapa laptop saya tidak bisa buka email kantor?"

> *Petugas helpdesk IT di banyak perusahaan setiap hari menerima laporan serupa dari karyawan: "tidak bisa akses sistem internal", "muncul peringatan sertifikat tidak aman", atau "tidak bisa kirim email kantor dari rumah". Bab ini melatih peserta didik berpikir seperti petugas helpdesk yang memahami keseluruhan infrastruktur di baliknya: apakah VPN sudah tersambung, apakah DNS mengarah ke server yang benar, apakah sertifikat CA sudah terinstal di perangkat, dan apakah kredensial email sudah benar. Kemampuan menguji dan mendiagnosis dari sisi pengguna akhir inilah yang membedakan teknisi jaringan berpengalaman dari yang hanya bisa mengonfigurasi server tanpa memverifikasi hasilnya dari sudut pandang pemakai.*

Aktivitas Pembelajaran

1. Praktik pembuatan akun lokal dan konfigurasi hak sudo pada client.

2. Praktik konfigurasi dan pengujian koneksi VPN dari client ke gateway.

3. Praktik instalasi sertifikat Root CA dan pengujian akses web internal tanpa galat SSL.

4. Praktik konfigurasi dan pengujian mail client menggunakan akun LDAP.

5. Presentasi kelompok: setiap kelompok mendemonstrasikan seluruh alur pengujian end-to-end di depan kelas.

Rangkuman

Bab ini menutup rangkaian pembangunan infrastruktur ITNSA.ID dengan memverifikasi seluruh layanan dari sudut pandang pengguna akhir: akun lokal dengan hak sudo, koneksi VPN yang aktif otomatis, akses web internal maupun publik dengan sertifikat SSL yang diterima, serta klien email yang terautentikasi melalui LDAP. Tahap pengujian sisi klien ini penting karena menjadi bukti nyata bahwa seluruh konfigurasi teknis pada Bab 2 hingga Bab 8 benar-benar berfungsi dan dapat digunakan sebagaimana mestinya oleh pengguna akhir.

Soal Latihan / Refleksi

1. Jelaskan mengapa pengujian sisi klien penting dilakukan setelah seluruh layanan server dikonfigurasi.

2. Jelaskan langkah agar koneksi VPN pada client aktif secara otomatis setiap kali laptop dinyalakan.

3. Jelaskan mengapa sertifikat Root CA perlu diinstal secara manual pada client agar tidak muncul peringatan SSL.

4. Sebutkan parameter yang perlu dikonfigurasi pada aplikasi mail client agar dapat terhubung ke mail server ITNSA.ID.

BAB 10 — Virtualisasi Server dengan Proxmox VE

Tujuan Pembelajaran

- Peserta didik mampu menjelaskan konsep virtualisasi server beserta jenis-jenis hypervisor.
- Peserta didik mampu menginstal dan mengonfigurasi Proxmox Virtual Environment (VE) 8.0.
- Peserta didik mampu membuat dan mengelola mesin virtual (VM) berbasis KVM serta container (LXC) pada Proxmox VE.
- Peserta didik mampu mengelola user, permission, backup/restore, firewall, dan proteksi brute force pada Proxmox VE.

10.1 Konsep Dasar Virtualisasi

Virtualisasi adalah teknologi yang memungkinkan satu perangkat keras fisik (physical server) menjalankan beberapa sistem operasi secara bersamaan dan terisolasi satu sama lain, seolah-olah setiap sistem operasi tersebut berjalan pada perangkat keras yang terpisah. Pada delapan bab sebelumnya, peserta didik telah mempraktikkan seluruh layanan ITNSA.ID (DNS, LDAP, PKI, Mail, Web, Gateway) menggunakan mesin virtual pada VMware Workstation. Bab ini memperkenalkan Proxmox VE sebagai platform virtualisasi kelas enterprise berbasis Linux yang open-source, yang lazim digunakan sebagai pengganti VMware ESXi/vSphere di banyak perusahaan maupun institusi pendidikan karena tidak memerlukan lisensi berbayar untuk fitur intinya.

10.2 Jenis Hypervisor

Perangkat lunak yang mengelola dan menjalankan mesin virtual disebut hypervisor. Terdapat dua jenis hypervisor berdasarkan cara instalasinya.

Jenis Hypervisor	Karakteristik	Contoh
Tipe 1 (Bare-Metal)	Terinstal langsung di atas perangkat keras tanpa memerlukan sistem operasi host, sehingga overhead lebih rendah dan performa lebih tinggi — cocok untuk lingkungan produksi/server	Proxmox VE, VMware ESXi, Microsoft Hyper-V (mode server)
Tipe 2 (Hosted)	Terinstal sebagai aplikasi di atas sistem operasi host yang sudah ada, lebih mudah dipakai untuk uji coba/laboratorium personal namun overhead lebih tinggi	Oracle VirtualBox, VMware Workstation

Pada praktik pembelajaran di laboratorium sekolah dengan keterbatasan jumlah perangkat keras fisik, Proxmox VE (hypervisor tipe 1) umumnya dijalankan sebagai mesin virtual bertingkat (nested virtualization) di atas Oracle VirtualBox (hypervisor tipe 2) yang sudah terinstal pada laptop/PC peserta didik — teknik inilah yang digunakan sepanjang bab ini, mengikuti pendekatan pada buku Panduan Praktikum Administrasi Sistem Jaringan – Proxmox VE 8.0.

Sumber: I Putu Hariyadi, Panduan Praktikum Administrasi Sistem Jaringan – Proxmox VE 8.0 (2023), Bab I.

10.3 Full Virtualization (KVM) vs Container-Based Virtualization (LXC)

Proxmox VE mendukung dua model virtualisasi sekaligus dalam satu platform, dengan karakteristik dan kegunaan yang berbeda.

Model	Cara Kerja	Kelebihan	Kekurangan
KVM (Kernel-based Virtual Machine)	Virtualisasi penuh; setiap VM menjalankan kernel sistem operasinya sendiri, terisolasi penuh dari host	Dapat menjalankan sistem operasi apa pun (Windows, Linux, BSD, bahkan RouterOS/MikroTik CHR); isolasi paling kuat	Overhead resource lebih besar dibanding container
LXC (Linux Container)	Virtualisasi tingkat sistem operasi; seluruh container berbagi kernel Linux yang sama dengan host	Sangat ringan, waktu boot cepat, efisien dalam pemakaian RAM/CPU	Hanya dapat menjalankan sistem operasi Linux, isolasi tidak sekuat KVM

Pemilihan model bergantung pada kebutuhan: MikroTik Cloud Hosted Router (CHR) yang membutuhkan kernel RouterOS sendiri wajib dijalankan sebagai VM KVM, sedangkan layanan berbasis Linux seperti web server atau streaming server dapat dijalankan lebih efisien sebagai LXC container.

10.4 Instalasi Proxmox VE 8.0

Instalasi Proxmox VE dilakukan menggunakan installer ISO resmi yang diunduh dari situs proxmox.com. Pada skenario laboratorium bertingkat (nested), Proxmox VE diinstalasi sebagai guest VM pada Oracle VirtualBox dengan spesifikasi minimal RAM 3072 MB, hard disk 40 GB, serta fitur nested virtualization (VT-x/AMD-V) diaktifkan pada pengaturan prosesor VM agar Proxmox VE mampu menjalankan VM KVM di dalamnya.

1. Buat VM baru pada Oracle VirtualBox dengan tipe Other/Unknown (64-bit), alokasikan RAM 3072 MB dan hard disk 40 GB (dynamically allocated).

2. Aktifkan opsi "Enable Nested VT-x/AMD-V" pada menu Settings > System > Processor VM tersebut.

3. Pasang (mount) berkas ISO installer Proxmox VE 8.0 pada optical drive VM, lalu jalankan VM untuk memulai instalasi.

4. Ikuti wizard instalasi: setujui EULA, pilih target hard disk, tentukan zona waktu dan keyboard layout, isikan password root serta alamat email admin.

5. Konfigurasikan pengalamatan jaringan: hostname (misalnya pve.itnsa.id), alamat IP statis (contoh 192.168.169.1/24), gateway, dan DNS server.

6. Setelah instalasi selesai dan server reboot, akses web interface administrasi Proxmox melalui browser pada alamat https://192.168.169.1:8006.

Sumber: I Putu Hariyadi, Panduan Praktikum Administrasi Sistem Jaringan – Proxmox VE 8.0 (2023), Bab I bagian B.

10.5 Arsitektur dan Komponen Utama Proxmox VE

Komponen	Fungsi
Web GUI (pveproxy)	Antarmuka administrasi berbasis web (port 8006) untuk mengelola seluruh node, VM, container, storage, dan jaringan
pmxcfs	Proxmox Cluster File System — sistem berkas terdistribusi berbasis database yang menyimpan seluruh konfigurasi node/VM/container dan mereplikasinya secara real-time ke seluruh anggota cluster
Node	Satu unit fisik/virtual yang menjalankan Proxmox VE; satu instalasi mandiri disebut node tunggal, beberapa node yang digabung disebut cluster (dibahas pada Bab 11)

Komponen	Fungsi
Storage	Lokasi penyimpanan disk image VM, container, ISO, dan backup — dapat berupa local storage maupun shared storage (NFS, iSCSI, Ceph)
vzdump	Utilitas bawaan untuk melakukan backup konsisten terhadap VM/container yang sedang berjalan

10.6 Menonaktifkan Notifikasi "No Valid Subscription"

Sebagai perangkat lunak open-source dengan model bisnis dukungan berbayar (subscription), Proxmox VE tanpa lisensi berbayar akan menampilkan pesan peringatan "No valid subscription" setiap kali admin login ke web interface. Pesan ini bersifat kosmetik dan tidak memengaruhi fungsi sistem, namun dapat dinonaktifkan melalui akses SSH dengan memodifikasi berkas JavaScript antarmuka web.

```
root@pve:~# sed -Ezi.bak \
  "s/(Ext.Msg.show\(\{\s+title:\s+gettext\('No\s+valid\s+sub)/void\(\{ \/\/\1/g" \
  /usr/share/javascript/proxmox-widget-toolkit/proxmoxlib.js
root@pve:~# systemctl restart pveproxy.service
```

Sumber: I Putu Hariyadi, Panduan Praktikum Administrasi Sistem Jaringan – Proxmox VE 8.0 (2023), Bab II.

10.7 Manajemen Repository

Secara default, Proxmox VE dikonfigurasi menggunakan repository enterprise yang memerlukan subscription aktif untuk dapat diakses. Agar perintah apt update/upgrade tidak menghasilkan galat pada instalasi tanpa subscription, repository enterprise perlu dinonaktifkan dan digantikan dengan repository no-subscription yang tersedia gratis untuk keperluan pembelajaran dan pengujian.

```
# Menonaktifkan repository enterprise
root@pve:~# echo "# deb https://enterprise.proxmox.com/debian/pve bookworm pve-enterprise" \
  > /etc/apt/sources.list.d/pve-enterprise.list

# Menambahkan repository no-subscription
root@pve:~# echo "deb http://download.proxmox.com/debian/pve bookworm pve-no-subscription" \
  > /etc/apt/sources.list.d/pve-no-subscription.list
root@pve:~# apt update
```

Sumber: I Putu Hariyadi, Panduan Praktikum Administrasi Sistem Jaringan – Proxmox VE 8.0 (2023), Bab III.

10.8 Membuat Mesin Virtual (VM) Berbasis KVM

Sebagai penerapan konkret, MikroTik Cloud Hosted Router (CHR) — versi RouterOS yang didistribusikan dalam bentuk disk image untuk platform virtualisasi — dapat dijalankan sebagai VM KVM pada Proxmox VE untuk berperan sebagai internet gateway pada topologi laboratorium, menggantikan fungsi gateway fisik pada Bab 8. Menurut wiki resmi MikroTik, CHR mendukung arsitektur x86 64-bit dan dapat dijalankan pada berbagai hypervisor termasuk Proxmox, dengan fitur RouterOS lengkap aktif secara default.

Kebutuhan Sistem	Spesifikasi Minimum CHR
CPU	64-bit dengan dukungan virtualisasi (VT-x/AMD-V)
Memori (RAM)	128 MB atau lebih untuk satu instance CHR
Hard disk (virtual drive)	128 MB, dengan ukuran disk image yang didukung hingga 16 GB

Kebutuhan Sistem	Spesifikasi Minimum CHR
Koneksi Internet pada host Proxmox	Diperlukan saat instalasi untuk mengunduh image CHR dan paket unzip

Sumber: I Putu Hariyadi, Panduan Praktikum Administrasi Sistem Jaringan – Proxmox VE 8.0 (2023), Bab V bagian Pendahuluan, mengutip wiki MikroTik.

1. Unggah berkas image CHR (.vhdx/.img/.qcow2) ke storage Proxmox menggunakan WinSCP atau perintah wget langsung dari console Proxmox.

2. Buat VM baru melalui menu Create VM pada web interface, tentukan VM ID, nama, dan alokasikan CPU/RAM secukupnya (misalnya 1 vCPU, 256–512 MB RAM, karena RouterOS ringan).

3. Impor disk image CHR ke VM yang baru dibuat menggunakan perintah qm importdisk pada console Proxmox.

4. Pasang (attach) disk hasil impor ke VM, atur boot order agar boot dari disk tersebut, lalu tentukan network adapter yang terhubung ke bridge vmbr0/vmbr1 sesuai topologi.

5. Jalankan (start) VM, buka console-nya melalui web interface, lalu lakukan konfigurasi dasar RouterOS (alamat IP, NAT, DHCP) seperti pada Bab 8.

```
root@pve:~# qm create 201 --name mikrotik-chr --memory 512 --cores 1 --net0
virtio,bridge=vmbr0
root@pve:~# qm importdisk 201 chr-7.15.img local-lvm
root@pve:~# qm set 201 --scsihw virtio-scsi-pci --scsi0 local-lvm:vm-201-disk-0
root@pve:~# qm set 201 --boot order=scsi0
root@pve:~# qm start 201
```

Sumber: I Putu Hariyadi, Panduan Praktikum Administrasi Sistem Jaringan – Proxmox VE 8.0 (2023), Bab IV dan Bab V.

Setelah VM berjalan, konfigurasi dasar RouterOS dilakukan melalui menu Console pada web interface Proxmox, tanpa memerlukan aplikasi tambahan seperti Winbox pada tahap awal. Login pertama kali menggunakan username admin dengan password kosong, kemudian sistem akan meminta penggantian password baru sebagai praktik keamanan dasar.

1. Login ke console CHR dengan username admin dan password dikosongkan (tekan Enter), lalu ketik n saat ditanya apakah ingin melihat lisensi perangkat lunak.

2. Buat password baru sesuai kebijakan keamanan sekolah saat diminta pada prompt new password>.

3. Tampilkan daftar interface yang tersedia menggunakan perintah interface print untuk memastikan ether1 dan ether2 telah terdeteksi VM.

4. Atur alamat IP statis pada interface yang menghadap ke jaringan internal (misalnya ether2) menggunakan perintah ip address add address=192.168.169.254/24 interface=ether2.

5. Aktifkan fitur DNS server bawaan RouterOS dan Network Address Translation (NAT) menggunakan ip dns set dan ip firewall nat add agar client internal dapat mengakses internet melalui CHR ini, mengikuti prinsip yang sama seperti gateway ITNSA.ID pada Bab 8.

6. Verifikasi konektivitas ke internet menggunakan perintah ping dari dalam CHR, misalnya ping detik.com, lalu hentikan dengan CTRL+C.

10.9 Membuat Container (LXC)

Untuk layanan berbasis Linux yang lebih ringan, Proxmox VE menyediakan template container siap pakai (misalnya CentOS 9 Stream, Debian, Ubuntu) yang dapat diunduh langsung dari repository resmi dan dijalankan sebagai LXC container dalam hitungan detik.

```
root@pve:~# pveam update
root@pve:~# pveam available | grep centos
root@pve:~# pveam download local centos-9-stream-default_20221109_amd64.tar.xz
root@pve:~# pct create 301 local:vztmpl/centos-9-stream-default_20221109_amd64.tar.xz \
  --hostname streaming-01 --memory 512 --cores 1 \
  --net0 name=eth0,bridge=vmbr0,ip=192.168.169.5/24,gw=192.168.169.1
root@pve:~# pct start 301
root@pve:~# pct enter 301
```

Setelah container aktif, langkah lanjutan seperti instalasi OpenSSH, aktivasi service sshd, dan pengaturan PermitRootLogin dilakukan persis seperti administrasi server Linux biasa (bandingkan dengan konfigurasi Postfix/Dovecot pada Bab 5), sehingga kompetensi administrasi Linux yang telah dipelajari pada bab-bab sebelumnya tetap relevan sepenuhnya di lingkungan container.

Sumber: I Putu Hariyadi, Panduan Praktikum Administrasi Sistem Jaringan – Proxmox VE 8.0 (2023), Bab VI dan Bab XI.

10.10 Manajemen User dan Permission Berbasis Role

Proxmox VE mendukung berbagai sumber otentikasi pengguna, meliputi Linux PAM, Proxmox VE Authentication Server, LDAP, dan Microsoft Active Directory — termasuk kemungkinan integrasi dengan server LDAP ITNSA.ID yang telah dibangun pada Bab 3. Akses granular diatur menggunakan sistem role, yaitu kumpulan hak akses yang telah didefinisikan, kemudian dikaitkan (assign) ke user atau grup pada path/objek tertentu (misalnya VM tertentu, storage tertentu, atau seluruh datacenter).

Role	Cakupan Hak Akses
Administrator	Seluruh hak akses tanpa batasan
PVEAdmin	Hampir seluruh hak akses, kecuali mengubah pengaturan sistem inti (Sys.PowerMgmt, Sys.Modify)
PVEAuditor	Hanya akses baca (read only), cocok untuk keperluan audit
PVEVMAdmin	Mengelola VM secara penuh (start, stop, konfigurasi, console)
PVEVMUser	Hak terbatas pada VM: melihat, backup, mengubah CDROM, akses console, power management
NoAccess	Tidak memiliki hak akses sama sekali, digunakan untuk secara eksplisit melarang akses

```
root@pve:~# pveum user add teknisi01@pve --password
root@pve:~# pveum aclmod /vms/201 --user teknisi01@pve --role PVEVMUser
root@pve:~# pveum user list
```

Sumber: I Putu Hariyadi, Panduan Praktikum Administrasi Sistem Jaringan – Proxmox VE 8.0 (2023), Bab VII.

10.11 Backup dan Restore

Utilitas bawaan vzdump memungkinkan admin membuat backup konsisten dari VM atau container yang sedang berjalan tanpa perlu mematikannya (live backup), disimpan dalam format arsip terkompresi yang dapat direstorasi kembali kapan pun diperlukan — kemampuan ini menjadi jaring pengaman (safety net) sebelum melakukan perubahan konfigurasi berisiko pada layanan produksi ITNSA.ID.

```
root@pve:~# vzdump 201 --mode snapshot --compress zstd --storage local
root@pve:~# ls /var/lib/vz/dump/
```

```
# Restore VM dari file backup
root@pve:~# qmrestore /var/lib/vz/dump/vzdump-qemu-201-*.vma.zst 202
```

Sumber: I Putu Hariyadi, Panduan Praktikum Administrasi Sistem Jaringan – Proxmox VE 8.0 (2023), Bab VIII.

10.12 Firewall dan Proteksi Brute Force (Fail2ban)

Proxmox VE menyediakan firewall terintegrasi yang dapat diterapkan pada tingkat datacenter, node, maupun VM/container secara individual, memungkinkan admin membatasi akses ke web interface administrasi (port 8006) maupun layanan pada VM/container tertentu. Sebagai lapisan proteksi tambahan terhadap percobaan login paksa (brute force) ke SSH maupun web interface, Fail2ban dapat dipasang untuk secara otomatis memblokir alamat IP yang berulang kali gagal login dalam rentang waktu tertentu.

```
root@pve:~# apt install fail2ban -y

# /etc/fail2ban/jail.local
[proxmox]
enabled = true
port = https,http,8006
filter = proxmox
logpath = /var/log/daemon.log
maxretry = 3
bantime = 3600
findtime = 600

root@pve:~# systemctl restart fail2ban
root@pve:~# fail2ban-client status proxmox
```

Sumber: I Putu Hariyadi, Panduan Praktikum Administrasi Sistem Jaringan – Proxmox VE 8.0 (2023), Bab IX dan Bab X.

10.13 Jaringan Virtual pada Proxmox VE

Proxmox VE menggunakan model jaringan berbasis bridge Linux (vmbr) yang berfungsi layaknya switch virtual: seluruh VM/container yang terhubung ke bridge yang sama seolah-olah terhubung ke kabel jaringan pada switch fisik yang sama. Secara default, instalasi Proxmox membentuk satu bridge (vmbr0) yang terhubung ke NIC fisik pertama, namun admin dapat membuat bridge tambahan untuk memisahkan segmen jaringan, misalnya memisahkan jaringan internal, DMZ, dan jaringan manajemen ITNSA.ID sebagaimana dirancang pada Bab 1.

Konsep Jaringan	Fungsi pada Proxmox VE
Linux Bridge (vmbr)	Switch virtual software yang menghubungkan VM/container satu sama lain maupun ke NIC fisik host
VLAN-aware Bridge	Satu bridge yang mampu memisahkan trafik banyak VLAN sekaligus (IEEE 802.1Q), setiap VM/container tinggal diberi tag VLAN ID tanpa perlu bridge terpisah per VLAN
Bonding/Link Aggregation	Menggabungkan beberapa NIC fisik menjadi satu interface logis untuk redundansi dan/atau penambahan throughput — penting bagi jaringan cluster (Bab 11) yang sensitif terhadap latensi
Firewall per-VM	Aturan firewall dapat diterapkan langsung pada level interface virtual VM/container, melengkapi firewall pada level node maupun datacenter

Sebagai contoh penerapan pada ITNSA.ID, admin dapat membuat vmbr0 untuk jaringan manajemen Proxmox, vmbr1 dengan VLAN-aware untuk memisahkan VLAN internal (int-srv, LDAP) dan VLAN DMZ (mail,

web-01, web-02) hanya dengan satu NIC fisik, serta vmbr2 khusus untuk trafik cluster Corosync yang dipisahkan dari trafik VM agar tidak mengganggu stabilitas quorum saat trafik VM sedang tinggi.

```
# /etc/network/interfaces (cuplikan konfigurasi VLAN-aware bridge)
auto vmbr1
iface vmbr1 inet manual
    bridge-ports eth1
    bridge-stp off
    bridge-fd 0
    bridge-vlan-aware yes
    bridge-vids 10,20,30
```

10.14 Manajemen Storage pada Proxmox VE

Pemilihan jenis storage memengaruhi fitur apa saja yang dapat dimanfaatkan pada suatu VM/container, terutama terkait snapshot, live migration, dan HA (Bab 11). Proxmox VE mendukung berbagai backend storage yang dapat dikombinasikan sesuai kebutuhan.

Jenis Storage	Lokal/Bersama	Mendukung Snapshot	Cocok untuk HA/Live Migration
Directory (local)	Lokal	Tergantung format file (qcow2: ya)	Tidak (kecuali direplikasi)
LVM (thick)	Lokal/iSCSI	Tidak	Ya jika di atas shared iSCSI
LVM-Thin	Lokal	Ya	Tidak (thin pool tidak shareable)
ZFS	Lokal	Ya	Tidak (kecuali dengan replikasi ZFS, lihat 11.13)
NFS	Bersama (network)	Tergantung format file	Ya
iSCSI	Bersama (network)	Tidak (perlu LVM di atasnya)	Ya
Ceph RBD	Bersama (terdistribusi)	Ya	Ya — pilihan terbaik untuk cluster HA skala menengah–besar

Untuk laboratorium pembelajaran dengan keterbatasan perangkat keras, NFS menjadi pilihan paling praktis sebagai shared storage karena dapat dibangun dari satu container Linux biasa (seperti dipraktikkan pada Bab 11), sedangkan Ceph — meski merupakan pilihan terbaik untuk produksi karena tidak memiliki single point of failure pada storage-nya sendiri — memerlukan minimal tiga node dengan disk khusus dan spesifikasi jaringan yang lebih tinggi.

10.15 Proxmox Backup Server (PBS)

Selain utilitas vzdump bawaan (bagian 10.11) yang menyimpan backup sebagai arsip pada storage biasa, Proxmox menyediakan produk terpisah bernama Proxmox Backup Server (PBS) yang dirancang khusus untuk kebutuhan backup skala enterprise. PBS berjalan sebagai instalasi mandiri (dapat pula dijalankan sebagai VM/container pada node Proxmox VE yang sama) dan menawarkan kemampuan deduplikasi data tingkat blok, backup incremental yang sangat efisien, enkripsi sisi klien (client-side encryption), serta dapat menjadi target backup dari banyak cluster Proxmox VE sekaligus.

Aspek	vzdump (bawaan PVE)	Proxmox Backup Server (PBS)
Deduplikasi	Tidak	Ya, tingkat blok (chunk-based)
Backup Incremental	Terbatas	Ya, sangat efisien untuk VM besar
Enkripsi sisi klien	Tidak	Ya
Instalasi	Terintegrasi otomatis	Server terpisah (dedicated)

Untuk infrastruktur ITNSA.ID skala kecil-menengah pada lingkungan pembelajaran, vzdump sudah mencukupi; namun peserta didik perlu memahami keberadaan PBS sebagai solusi tingkat lanjut yang akan mereka jumpai di lingkungan kerja dengan kebutuhan retensi backup jangka panjang dan volume data lebih besar.

10.16 Monitoring dan Notifikasi

Web interface Proxmox VE menyediakan grafik penggunaan CPU, RAM, disk I/O, dan trafik jaringan secara real-time maupun historis untuk setiap node, VM, dan container, memudahkan admin memantau kapasitas dan merencanakan penambahan sumber daya (capacity planning). Selain itu, Proxmox dapat dikonfigurasi mengirim notifikasi otomatis melalui email atau webhook ketika terjadi kegagalan backup, peringatan penggunaan storage mendekati penuh, atau peristiwa penting lain pada cluster — termasuk peristiwa HA seperti fencing dan failover yang akan dipelajari pada Bab 11.

```
root@pve:~# pvesh set /cluster/notifications/endpoints/sendmail/admin-mail \
  --mailto admin@itnsa.id
```

10.17 Template VM dan Cloud-Init untuk Provisioning Cepat

Untuk mempercepat penyediaan (provisioning) banyak VM sejenis — misalnya beberapa VM Linux dengan konfigurasi dasar yang sama sebagai bahan praktikum siswa — Proxmox VE mendukung mekanisme template dan clone. Sebuah VM yang telah diinstal dan dikonfigurasi dasar dapat dijadikan template, kemudian di-clone berkali-kali dalam hitungan detik tanpa harus mengulang proses instalasi sistem operasi dari awal. Dipadukan dengan Cloud-Init (standar industri untuk konfigurasi otomatis VM saat boot pertama), admin dapat menentukan hostname, alamat IP, dan kunci SSH secara otomatis untuk setiap hasil clone tanpa perlu masuk ke console VM satu per satu.

```
root@pve:~# qm template 9000
root@pve:~# qm clone 9000 301 --name web-03 --full
root@pve:~# qm set 301 --ipconfig0 ip=10.10.20.13/24,gw=10.10.20.1
root@pve:~# qm set 301 --sshkeys ~/.ssh/authorized_keys
root@pve:~# qm start 301
```

Teknik ini sangat relevan bila ITNSA.ID hendak menambah kapasitas web server (misalnya web-03 sebagai tambahan backend HAProxy pada Bab 6) tanpa harus mengulang instalasi Nginx dari nol — cukup clone dari template yang sudah memiliki Nginx terpasang, lalu jalankan playbook Ansible (Bab 7) untuk menyesuaikan konfigurasi spesifik virtual host-nya.

10.18 Integrasi Proxmox VE dengan Ansible

Melanjutkan otomasi pada Bab 7, Ansible juga dapat mengelola siklus hidup VM/container pada Proxmox VE itu sendiri (bukan hanya mengonfigurasi isi VM), menggunakan modul community.general.proxmox_kvm

dan sejenisnya. Dengan demikian, seluruh proses mulai dari pembuatan VM baru, instalasi paket, hingga deployment layanan dapat didefinisikan sebagai satu playbook tunggal yang dapat dijalankan berulang secara idempoten.

```
# playbook-provision-vm.yml (dijalankan dari control node Ansible)
---
- name: Membuat VM baru di Proxmox secara otomatis
  hosts: localhost
  tasks:
    - name: Clone VM dari template
      community.general.proxmox_kvm:
        api_host: pve1.itnsa.id
        api_user: root@pam
        api_password: "{{ pve_password }}"
        clone: 9000
        newid: 302
        name: web-04
        node: pve1
        state: present
```

Pendekatan Infrastructure as Code (IaC) semacam ini — mendefinisikan seluruh infrastruktur, termasuk keberadaan VM itu sendiri, sebagai berkas konfigurasi yang dapat diulang dan diaudit — merupakan praktik standar tim DevOps/Platform Engineering modern, dan menjadi penutup yang tepat bagi rangkaian pembelajaran otomasi yang dimulai pada Bab 7.

10.19 Proxmox VE Dibandingkan Hypervisor Tipe 2 yang Telah Dipakai

Sepanjang Bab 2 hingga Bab 9, seluruh praktik ITNSA.ID dijalankan di atas VMware Workstation (hypervisor tipe 2) pada laptop masing-masing peserta didik. Tabel berikut merangkum perbedaan mendasar yang menjelaskan mengapa lingkungan produksi sesungguhnya beralih ke Proxmox VE (hypervisor tipe 1).

Aspek	VMware Workstation (Tipe 2)	Proxmox VE (Tipe 1)
Instalasi	Aplikasi di atas OS host (Windows/Linux desktop)	Sistem operasi tersendiri, terinstal langsung di perangkat keras server
Akses jarak jauh	Terbatas pada mesin tempat aplikasi terinstal	Web interface dapat diakses dari mana saja melalui jaringan
Cluster & HA	Tidak mendukung	Mendukung penuh (Bab 11)
Penggunaan tipikal	Laboratorium/pembelajaran pribadi, pengembangan	Produksi, data center, layanan pelanggan
Lisensi	Berbayar untuk versi Pro	Gratis untuk fitur inti (community/no-subscription)

Pemahaman ini penting agar peserta didik menyadari bahwa VMware Workstation yang dipakai pada Bab 2–9 adalah pilihan tepat untuk tahap belajar dan bereksperimen secara individual, sedangkan Proxmox VE pada Bab 10–11 merepresentasikan bagaimana infrastruktur yang sama akan benar-benar dioperasikan di lingkungan kerja profesional maupun data center produksi ITNSA.ID yang sesungguhnya.

10.20 Instalasi Otomatis (Automated Installation) Proxmox VE 8.0

Untuk kebutuhan penyediaan banyak node sekaligus — misalnya laboratorium sekolah yang hendak memasang Proxmox VE pada belasan unit komputer secara seragam — Proxmox VE 8.0 memperkenalkan

fitur Automated Installation berbasis answer file berformat TOML. Berkas ini mendefinisikan seluruh jawaban yang biasanya diisi manual saat instalasi (partisi disk, hostname, password root, konfigurasi jaringan), sehingga instalasi dapat berjalan tanpa interaksi manusia sama sekali (unattended installation).

```
# answer.toml (contoh ringkas)
[global]
keyboard = "id"
country = "id"
fqdn = "pve1.itnsa.id"
mailto = "admin@itnsa.id"
root_password = "PasswordSangatKuat123!"

[network]
source = "from-answer"
cidr = "192.168.169.1/24"
gateway = "192.168.169.254"
dns = "192.168.169.1"

[disk-setup]
filesystem = "ext4"
disk_list = ["sda"]
```

Berkas answer.toml ini kemudian disertakan pada proses pembuatan ISO installer khusus menggunakan utilitas proxmox-auto-install-assistant, sehingga tim IT ITNSA.ID cukup mem-boot setiap unit komputer baru menggunakan satu ISO yang sama untuk mendapatkan node Proxmox VE dengan konfigurasi dasar yang identik dan konsisten, mengurangi risiko kesalahan konfigurasi manual berulang.

Sumber: Dokumentasi resmi Proxmox VE, bagian Automated Installation (pve.proxmox.com/pve-docs).

10.21 Estimasi Kebutuhan Sumber Daya Layanan ITNSA.ID

Sebagai persiapan sebelum menempatkan seluruh layanan ITNSA.ID di atas Proxmox VE, berikut estimasi kebutuhan sumber daya tipikal setiap layanan yang telah dibangun pada Bab 2 hingga Bab 9, sebagai bahan perencanaan kapasitas node.

VM/Container	vCPU	RAM	Disk	Jenis Virtualisasi yang Disarankan
int-srv (DNS+LDAP+CA)	2	2 GB	20 GB	LXC (seluruhnya layanan Linux ringan)
mail (Postfix+Dovecot)	2	4 GB	40 GB	LXC atau KVM tergantung volume email
web-01 / web-02	1	2 GB	20 GB	LXC (Nginx ringan)
gateway (nftables+WireGuard)	2	2 GB	20 GB	KVM (mengelola interface jaringan langsung)
mikrotik-chr (opsional, Bab 10)	1	512 MB	1 GB	KVM (memerlukan kernel RouterOS sendiri)

Total kebutuhan pada tabel di atas (sekitar 8 vCPU dan 10,5 GB RAM) menjadi acuan dasar penerapan prinsip N+1 yang akan dibahas lebih lanjut pada bagian 11.16 — sehingga kapasitas setiap node cluster idealnya direncanakan melebihi sekadar sepertiga dari total kebutuhan tersebut.

10.22 Snapshot dan Rollback VM

Selain backup penuh melalui vzdump, Proxmox VE menyediakan fitur snapshot yang merekam kondisi VM/container pada suatu titik waktu tertentu (termasuk isi disk dan, untuk VM KVM yang sedang berjalan, kondisi memori) tanpa perlu menyalin seluruh data seperti backup penuh. Snapshot sangat berguna sebagai "titik aman" sebelum melakukan perubahan konfigurasi berisiko, misalnya sebelum menerapkan pembaruan sistem operasi besar pada int-srv atau menguji perubahan konfigurasi Postfix yang belum pernah dicoba sebelumnya.

```
root@pve:~# qm snapshot 100 sebelum-update-besar --description "Sebelum apt dist-upgrade"
root@pve:~# qm listsnapshot 100

# Apabila update bermasalah, kembalikan VM ke kondisi snapshot
root@pve:~# qm rollback 100 sebelum-update-besar
```

Perlu dicatat bahwa snapshot bukan pengganti backup: snapshot umumnya disimpan pada storage yang sama dengan VM aslinya, sehingga apabila storage tersebut rusak total, snapshot ikut hilang bersama data aslinya. Snapshot berfungsi sebagai jaring pengaman jangka pendek untuk eksperimen konfigurasi, sedangkan vzdump/PBS (bagian 10.11 dan 10.15) tetap wajib dijalankan secara terjadwal sebagai perlindungan data jangka panjang.

10.23 Rubrik Penilaian Praktik Virtualisasi Proxmox VE

Rubrik berikut dapat digunakan guru untuk menilai kinerja peserta didik dalam menyelesaikan kotak Praktik pada bagian sebelumnya.

Komponen Penilaian	Indikator	Bobot
Instalasi dan konfigurasi dasar	Proxmox VE terinstal dengan benar, subscription notice dan repository dikonfigurasi sesuai standar no-subscription	20%
Pembuatan VM dan Container	VM KVM (MikroTik CHR) dan LXC container (CentOS) berhasil dibuat, berjalan, dan dapat diakses	30%
Manajemen user dan permission	Akun dengan role terbatas berhasil dibuat dan hak aksesnya terverifikasi sesuai ketentuan	15%
Backup dan restore	Proses backup dan restore berhasil dilakukan dan dibuktikan dengan data yang identik	20%
Keamanan dasar (firewall & Fail2ban)	Firewall dan Fail2ban aktif dan terbukti memblokir percobaan akses tidak sah	15%

10.24 Total Cost of Ownership: Proxmox VE vs Solusi Komersial

Sebagai pertimbangan tambahan yang relevan bagi peserta didik yang kelak akan mengambil keputusan teknis di dunia kerja, berikut ilustrasi sederhana perbandingan estimasi biaya kepemilikan (Total Cost of Ownership/TCO) antara Proxmox VE dan solusi hypervisor komersial untuk skenario tiga node server pada infrastruktur berskala kecil-menengah seperti ITNSA.ID.

Komponen Biaya	Proxmox VE (No-Subscription/Community)	Hypervisor Komersial (ilustrasi umum)
Lisensi hypervisor per node	Rp 0 (open-source)	Berbayar per CPU/node per tahun

Komponen Biaya	Proxmox VE (No-Subscription/Community)	Hypervisor Komersial (ilustrasi umum)
Lisensi manajemen terpusat (setara vCenter)	Termasuk (terintegrasi bawaan)	Umumnya terpisah dan berbayar
Fitur HA/cluster	Termasuk penuh	Umumnya memerlukan tingkatan lisensi lebih tinggi
Dukungan resmi (support subscription)	Opsional, dapat dibeli bila diperlukan	Umumnya wajib untuk mendapat pembaruan keamanan tercepat
Biaya pembelajaran (learning curve) bagi tim	Perlu familiarisasi CLI berbasis Debian/Linux	Bervariasi tergantung ekosistem vendor

Perbandingan ini bersifat ilustratif dan dapat berbeda tergantung kebijakan harga vendor maupun kebutuhan spesifik organisasi, namun secara umum menjelaskan mengapa banyak institusi pendidikan dan UKM (Usaha Kecil Menengah) di Indonesia mempertimbangkan Proxmox VE sebagai alternatif yang layak secara teknis maupun finansial dibandingkan solusi virtualisasi komersial, terutama untuk kebutuhan yang tidak memerlukan dukungan enterprise 24/7 dengan SLA sangat ketat.

10.25 Keterbatasan Virtualisasi Node Tunggal

Sebelum melanjutkan ke Bab 11, penting ditegaskan bahwa seluruh kemampuan Proxmox VE yang dipelajari pada bab ini — VM, container, backup, firewall — masih beroperasi pada satu node tunggal. Artinya, meski VM/container dapat dipulihkan dari backup (bagian 10.11) atau snapshot (bagian 10.22) setelah terjadi kegagalan, seluruh proses tersebut masih memerlukan intervensi manual admin dan diikuti downtime selama proses pemulihan berlangsung, yang bisa memakan waktu mulai dari beberapa menit hingga berjam-jam tergantung ukuran data dan kecepatan storage backup.

Keterbatasan inilah yang menjadi motivasi utama pembahasan Bab 11: dengan menggabungkan beberapa node Proxmox VE menjadi satu cluster ber-HA, proses pemulihan yang sebelumnya manual dan memakan waktu lama dapat diotomasi sepenuhnya oleh HA Manager, sehingga downtime akibat kegagalan perangkat keras dapat ditekan hingga hanya dalam hitungan menit tanpa memerlukan admin yang siap siaga menekan tombol restore secara manual di tengah malam.

🔧 Praktik: Membangun Laboratorium Virtualisasi Proxmox VE untuk ITNSA.ID

1. Instal Proxmox VE 8.0 sebagai nested VM pada Oracle VirtualBox mengikuti langkah pada bagian 10.4.

2. Nonaktifkan pesan "No valid subscription" dan alihkan repository ke jalur no-subscription sesuai bagian 10.6 dan 10.7.

3. Buat satu VM KVM menjalankan MikroTik CHR sebagai simulasi gateway ITNSA.ID, konfigurasikan alamat IP dasar dan uji konektivitas ke internet.

4. Buat satu LXC container CentOS 9 Stream, instal OpenSSH, lalu uji akses SSH dari client Windows menggunakan Putty.

5. Buat satu akun pengguna baru dengan role PVEVMUser yang hanya dapat mengelola VM CHR yang telah dibuat, verifikasi pembatasan hak aksesnya.

6. Lakukan backup terhadap container CentOS menggunakan vzdump, hapus container tersebut, lalu pulihkan kembali menggunakan hasil backup untuk membuktikan proses restore berhasil.

7. Aktifkan Fail2ban untuk melindungi web interface Proxmox, lalu simulasikan beberapa kali percobaan login gagal dan verifikasi IP penyerang diblokir menggunakan fail2ban-client status.

Contoh dalam Kehidupan Nyata: Proxmox VE sebagai fondasi data center modern

Banyak perusahaan menengah di Indonesia beralih dari VMware ke Proxmox VE dalam beberapa tahun terakhir karena tidak memerlukan biaya lisensi tahunan yang mahal, sementara fitur intinya (VM, container, backup, cluster) tetap setara untuk kebutuhan operasional sehari-hari. Sebuah data center modern jarang memiliki satu server fisik untuk satu fungsi — sebaliknya, satu server fisik berkapasitas besar dipecah menjadi puluhan VM/container menggunakan Proxmox, masing-masing menjalankan layanan berbeda (mirip seluruh layanan ITNSA.ID pada Bab 2–9 yang sesungguhnya dapat dijalankan sebagai container/VM terpisah pada satu Proxmox node). Kemampuan backup terjadwal dan restore cepat juga menjadi alasan utama institusi pendidikan mengadopsi virtualisasi: jika satu VM praktikum siswa rusak akibat kesalahan konfigurasi, instruktur cukup me-restore dari backup terakhir dalam hitungan menit, dibandingkan menginstal ulang dari awal.

10.26 Istilah Kunci Bab 10

Istilah	Penjelasan Singkat
Hypervisor	Perangkat lunak yang mengelola dan menjalankan mesin virtual
KVM	Kernel-based Virtual Machine, teknologi full virtualization pada Proxmox
LXC	Linux Container, teknologi virtualisasi tingkat sistem operasi yang berbagi kernel host
vmbr	Linux bridge, switch virtual penghubung VM/container pada Proxmox
pmxcfs	Sistem berkas cluster Proxmox yang menyimpan seluruh konfigurasi
vzdump	Utilitas backup bawaan Proxmox VE untuk VM dan container
Nested Virtualization	Menjalankan hypervisor di dalam VM pada hypervisor lain, dipakai untuk lab Proxmox di atas VirtualBox
Template	VM/container yang telah dikonfigurasi dasar dan siap digandakan (clone) berulang kali

Aktivitas Pembelajaran

1. Praktik instalasi Proxmox VE sebagai nested VM dan konfigurasi dasar pasca-instalasi (subscription, repository).
2. Praktik membuat VM KVM (MikroTik CHR) dan LXC container (CentOS) pada Proxmox VE.
3. Praktik manajemen user/permission, backup/restore, firewall, dan Fail2ban.
4. Diskusi kelompok: dalam kondisi seperti apa sebaiknya suatu layanan dijalankan sebagai VM KVM, dan dalam kondisi seperti apa lebih tepat dijalankan sebagai LXC container?

Rangkuman

Proxmox VE adalah platform virtualisasi bare-metal open-source yang mendukung dua model virtualisasi (KVM untuk full virtualization dan LXC untuk container), dilengkapi web interface administrasi, sistem file cluster terdistribusi (pmxcfs), manajemen user berbasis role, backup/restore terintegrasi (vzdump dan opsional Proxmox Backup Server), jaringan virtual berbasis Linux bridge dan VLAN, berbagai pilihan storage (Directory, LVM, ZFS, NFS, iSCSI, Ceph), template/clone untuk provisioning cepat, firewall, dan proteksi brute force melalui Fail2ban. Dibandingkan hypervisor tipe 2 seperti VMware Workstation yang dipakai pada Bab 2–9, Proxmox VE merepresentasikan bagaimana infrastruktur ITNSA.ID akan dioperasikan secara nyata di lingkungan produksi. Bab ini menjadi fondasi sebelum peserta didik mempelajari penggabungan beberapa node Proxmox menjadi cluster untuk mendukung High Availability pada Bab 11.

Soal Latihan / Refleksi

1. Jelaskan perbedaan hypervisor tipe 1 dan tipe 2, beserta posisi Proxmox VE di antara keduanya.

2. Jelaskan perbedaan mendasar antara virtualisasi KVM dan LXC, serta berikan satu contoh skenario penggunaan masing-masing.

3. Jelaskan mengapa repository enterprise perlu dinonaktifkan pada instalasi Proxmox VE tanpa subscription.

4. Jelaskan fungsi role PVEAuditor dan PVEVMUser pada sistem manajemen permission Proxmox VE.

5. Jelaskan manfaat fitur backup vzdump bagi kelangsungan operasional infrastruktur ITNSA.ID.

6. Jelaskan perbedaan fungsi vzdump dan Proxmox Backup Server (PBS), serta kapan sebaiknya PBS dipertimbangkan.

7. Jelaskan manfaat fitur template dan clone VM dalam mempercepat penyediaan (provisioning) server baru pada infrastruktur ITNSA.ID.

BAB 11 — High Availability Virtualization dengan Proxmox VE Cluster

Tujuan Pembelajaran

- Peserta didik mampu menjelaskan konsep High Availability (HA) pada tingkat infrastruktur virtualisasi, berbeda dengan HA tingkat aplikasi pada Bab 6.
- Peserta didik mampu menjelaskan arsitektur cluster Proxmox VE (Corosync, pmxcfs, quorum).
- Peserta didik mampu membangun cluster Proxmox VE beserta syarat shared storage-nya.
- Peserta didik mampu mengonfigurasi HA Manager, HA Group, fencing, dan menguji mekanisme failover.

11.1 High Availability pada Tingkat Infrastruktur vs Tingkat Aplikasi

Pada Bab 6, peserta didik telah membangun High Availability pada tingkat aplikasi menggunakan Keepalived dan HAProxy — pendekatan ini bekerja dengan menjalankan dua instance aplikasi (Nginx) secara paralel di dua server terpisah, kemudian menyediakan mekanisme failover di atasnya. Bab ini memperkenalkan pendekatan HA yang berbeda dan saling melengkapi, yaitu High Availability pada tingkat infrastruktur virtualisasi: alih-alih menduplikasi aplikasi secara manual, seluruh VM/container dikelola oleh cluster Proxmox VE yang secara otomatis memindahkan dan menjalankan ulang VM yang terdampak ke node lain apabila node tempatnya berjalan mengalami kegagalan perangkat keras.

Kedua pendekatan ini idealnya digabungkan pada infrastruktur produksi nyata: HA tingkat infrastruktur (Bab 11) memastikan VM tetap "hidup" meski server fisik tempatnya berjalan padam, sedangkan HA tingkat aplikasi (Bab 6) memastikan layanan tetap dapat diakses bahkan selama proses failover VM berlangsung, karena permintaan pengguna sudah dialihkan ke instance lain yang masih aktif.

Contoh dalam Kehidupan Nyata: Dua lapis HA pada dunia nyata: analogi maskapai penerbangan

> *HA tingkat infrastruktur dapat dianalogikan seperti maskapai penerbangan yang memiliki pesawat cadangan siap terbang apabila pesawat utama mengalami kerusakan teknis mendadak — penumpang tetap sampai tujuan meski harus berpindah pesawat dan mengalami sedikit keterlambatan (downtime singkat, sesuai sifat restart-based HA pada Proxmox). Sementara itu, HA tingkat aplikasi seperti pada Bab 6 dapat dianalogikan seperti maskapai yang memiliki banyak penerbangan paralel ke tujuan yang sama sepanjang hari — jika satu penerbangan dibatalkan, penumpang dapat langsung dialihkan ke penerbangan lain yang sudah berjalan tanpa harus menunggu pesawat pengganti disiapkan (near zero-downtime). Infrastruktur kelas enterprise umumnya menggabungkan keduanya untuk keandalan maksimal.*

11.2 Arsitektur Cluster Proxmox VE

Cluster Proxmox VE dibangun di atas beberapa komponen inti yang bekerja sama untuk menjaga konsistensi data dan mendeteksi kegagalan node secara real-time.

Komponen	Fungsi
Corosync	Lapisan komunikasi cluster (cluster messaging) yang menjaga seluruh node saling mengetahui status satu sama lain melalui pertukaran heartbeat secara berkala
pmxcfs	Proxmox Cluster File System — mereplikasi seluruh konfigurasi (VM, storage, user, firewall) secara real-time ke seluruh node anggota cluster, sehingga setiap node memiliki salinan konfigurasi yang identik
Votequorum	Mekanisme voting yang menentukan apakah cluster memiliki quorum (mayoritas suara) untuk dapat mengambil keputusan secara aman

Komponen	Fungsi
CRM (Cluster Resource Manager)	Mengambil keputusan HA pada tingkat cluster: memutuskan node mana yang menjalankan VM tertentu dan memindahkannya bila terjadi kegagalan
LRM (Local Resource Manager)	Berjalan pada setiap node, menjalankan perintah dari CRM secara lokal, misalnya start/stop/migrate VM pada node bersangkutan

Sebuah keunggulan arsitektural penting Proxmox VE dibandingkan solusi klaster lain adalah tidak adanya single point of failure pada sisi manajemen — setiap node dapat melakukan seluruh tugas administrasi cluster, sehingga meski satu node (termasuk yang sedang menjadi master CRM) mati, node lain dapat langsung mengambil alih peran tersebut tanpa memerlukan server manajemen terpisah.

Sumber: Proxmox VE Administration Guide, dokumentasi resmi pve.proxmox.com, bagian Cluster Manager dan High Availability.

11.3 Syarat Membangun Cluster HA

Berbeda dengan node Proxmox VE tunggal pada Bab 10 yang cukup dijalankan sendiri, HA memerlukan beberapa prasyarat teknis yang wajib dipenuhi agar dapat berfungsi secara aman dan andal.

- **Minimal tiga node,** agar cluster tetap memiliki quorum (mayoritas suara) meski satu node mati. Dengan dua node saja, kehilangan satu node berarti kehilangan mayoritas, sehingga cluster tidak dapat mengambil keputusan HA secara aman.
- **Shared storage,** karena live migration dan HA mengharuskan disk image VM dapat diakses secara bersamaan oleh seluruh node, bukan tersimpan lokal di satu node saja. Jenis shared storage yang didukung meliputi NFS, iSCSI, dan Ceph (sistem penyimpanan terdistribusi bawaan Proxmox).
- **Jaringan cluster yang stabil dan berlatensi rendah,** karena Corosync sangat sensitif terhadap keterlambatan jaringan — disarankan menggunakan Network Interface Card (NIC) khusus/terdedikasi untuk trafik cluster, terpisah dari trafik VM maupun storage.
- **Versi Proxmox VE yang sama** pada seluruh node anggota cluster.

Sumber: Proxmox VE Cluster Manager Documentation, pve.proxmox.com/pve-docs/chapter-pvecm.html.

11.4 Membangun Cluster: pvecm

Pembentukan cluster dilakukan melalui perintah pvecm pada salah satu node yang akan menjadi node pertama, kemudian node-node lain bergabung ke cluster tersebut.

```
# Di node pertama (pve1)
root@pve1:~# pvecm create itnsa-cluster
root@pve1:~# pvecm status

# Di node kedua dan ketiga (pve2, pve3)
root@pve2:~# pvecm add 192.168.169.1
root@pve3:~# pvecm add 192.168.169.1

# Verifikasi keanggotaan cluster dari node mana pun
root@pve1:~# pvecm nodes
```

Setelah bergabung, konfigurasi seluruh node akan segera tersinkronisasi secara otomatis melalui pmxcfs, dan web interface administrasi Proxmox pada node mana pun akan menampilkan seluruh node cluster beserta VM/container di dalamnya.

11.5 Quorum dan QDevice

Quorum adalah jumlah suara minimum yang harus tercapai agar cluster dapat mengambil keputusan secara aman, dengan rumus umum mayoritas sederhana: lebih dari separuh total node. Pada cluster tiga node,

quorum tercapai jika minimal dua node aktif dan saling terhubung; kehilangan satu node masih menyisakan quorum (2 dari 3), namun kehilangan dua node akan membuat cluster kehilangan quorum dan menjadi read-only sebagai mekanisme keselamatan untuk mencegah split-brain.

Pada cluster dua node, kehilangan salah satu node berarti node yang tersisa hanya memiliki satu suara dari total dua — tidak mencapai mayoritas, sehingga HA tidak dapat berfungsi dengan aman. Untuk mengatasi keterbatasan ini tanpa harus menyediakan node Proxmox penuh ketiga, Proxmox VE mendukung QDevice (Corosync QNet), yaitu perangkat ringan (bahkan dapat berupa Raspberry Pi atau mesin Linux kecil) yang hanya menyediakan satu suara tambahan sebagai penentu (tie-breaker), tanpa perlu menjalankan Proxmox VE maupun VM apa pun.

```
# Pada mesin QDevice (Linux biasa, bukan node Proxmox)
user@qdevice:~$ sudo apt install corosync-qnetd

# Pada salah satu node Proxmox
root@pve1:~# pvecm qdevice setup 192.168.169.10
root@pve1:~# pvecm status
```

Sumber: Proxmox VE Cluster Manager Documentation dan HomeLab Starter, "Proxmox VE Clustering for High Availability" (2026).

11.6 Fencing: Mencegah Split-Brain dengan Watchdog

Fencing adalah mekanisme untuk memastikan sebuah node yang dianggap gagal (tidak merespons) benar-benar telah berhenti beroperasi sebelum VM miliknya dijalankan ulang di node lain. Tanpa fencing, dapat terjadi skenario split-brain, yaitu VM yang sama berjalan secara bersamaan di dua node berbeda sehingga menyebabkan korupsi data — misalnya jika node yang "dianggap mati" sesungguhnya hanya kehilangan koneksi jaringan sesaat namun VM di dalamnya tetap berjalan.

Proxmox VE menerapkan fencing berbasis watchdog: setiap node yang mengaktifkan HA menjalankan watchdog timer yang harus terus-menerus "disegarkan" (reset) oleh HA manager selama node tersebut memiliki quorum. Apabila node kehilangan quorum atau proses HA stack-nya gagal, watchdog tidak lagi disegarkan sehingga timer habis dan secara paksa me-reboot node tersebut (self-fencing) sebelum node lain mengambil alih VM-nya — memastikan tidak ada dua salinan VM yang berjalan bersamaan.

Jenis Watchdog	Karakteristik
Softdog (perangkat lunak)	Modul kernel bawaan yang digunakan secara default; cukup untuk kebutuhan laboratorium/pembelajaran namun kurang andal pada skenario kegagalan tertentu (misalnya kernel hang total)
Hardware watchdog (IPMI/iLO/iDRAC)	Watchdog berbasis perangkat keras pada motherboard server, direkomendasikan untuk lingkungan produksi karena tetap dapat memaksa reboot meski kernel/OS mengalami hang total

Sumber: Proxmox VE High Availability Documentation, pve.proxmox.com/pve-docs/chapter-ha-manager.html; ProxmoxR Blog, "Proxmox High Availability (HA) Configuration Guide" (2026).

11.7 HA Manager: Menambahkan VM ke HA

Setelah cluster dan shared storage siap, VM/container dapat didaftarkan sebagai resource HA menggunakan perintah ha-manager, lengkap dengan kebijakan pemulihan (recovery policy) yang menentukan berapa kali percobaan restart dilakukan pada node yang sama sebelum VM dipindahkan (relocate) ke node lain.

```
root@pve1:~# ha-manager add vm:201 --state started --max_restart 3 --max_relocate 1
root@pve1:~# ha-manager status
quorum OK
master pve1 (active, Mon Nov 21 07:23:29 2016)
lrm pve1 (active, Mon Nov 21 07:23:22 2016)
lrm pve2 (active, Mon Nov 21 07:23:24 2016)
lrm pve3 (active, Mon Nov 21 07:23:22 2016)
service vm:201 (pve1, started)
```

Status resource HA dapat bernilai started (dijalankan dan dijaga tetap berjalan), stopped (dihentikan namun tetap dipantau), disabled (dinonaktifkan sepenuhnya dari pengelolaan HA), atau ignored (diabaikan sementara, biasanya untuk keperluan maintenance).

11.8 HA Group dan Prioritas Node

HA Group memungkinkan admin menentukan node mana saja yang boleh menjalankan suatu VM tertentu beserta urutan prioritasnya, berguna misalnya untuk memastikan VM mail server ITNSA.ID (Bab 5) lebih diprioritaskan berjalan pada node dengan spesifikasi perangkat keras terbaik.

```
root@pve1:~# ha-manager groupadd grup-prioritas-mail --nodes pve1:100,pve2:50,pve3:50
root@pve1:~# ha-manager set vm:201 --group grup-prioritas-mail
```

Parameter nofailback menentukan apakah VM akan otomatis kembali (fail back) ke node dengan prioritas tertinggi begitu node tersebut pulih, atau tetap berada di node tempatnya berjalan saat ini agar tidak terjadi downtime tambahan akibat perpindahan bolak-balik yang tidak perlu.

11.9 Live Migration vs Relocation

Proxmox VE menyediakan dua mekanisme perpindahan VM antar node yang berbeda konteks penggunaannya.

Mekanisme	Kapan Digunakan	Dampak pada Layanan
Live Migration	Perpindahan terencana (maintenance node, load balancing manual) saat kedua node dalam kondisi sehat	VM tetap berjalan tanpa downtime; memori dan state VM disalin secara bertahap ke node tujuan
Relocation (via HA failover)	Perpindahan tidak terencana akibat kegagalan node, dieksekusi otomatis oleh CRM setelah proses fencing selesai	VM di-restart di node tujuan (bukan dilanjutkan state-nya), sehingga terjadi downtime singkat sesuai waktu boot VM

Perbedaan ini menegaskan sifat penting HA Proxmox: bersifat restart-based, bukan zero-downtime murni. HA meminimalkan downtime menjadi seukuran waktu boot VM/container (umumnya 1–3 menit tergantung metode fencing dan waktu boot sistem operasi tamu), bukan menghilangkannya sepenuhnya — berbeda dengan HA tingkat aplikasi pada Bab 6 yang dapat mendekati zero-downtime karena banyak instance aplikasi berjalan paralel secara bersamaan.

```
# Live migration manual (node sehat, direncanakan)
root@pve1:~# qm migrate 201 pve2 --online
```

Sumber: HomeLab Starter, "Proxmox VE Clustering for High Availability" (2026); Proxmox VE Administration Guide.

11.11 Ceph sebagai Shared Storage Terintegrasi

Ceph adalah sistem penyimpanan terdistribusi (software-defined storage) yang terintegrasi langsung ke dalam Proxmox VE, memungkinkan disk lokal pada setiap node cluster digabungkan menjadi satu storage pool bersama tanpa memerlukan perangkat NAS/SAN eksternal. Data pada Ceph secara otomatis direplikasi ke beberapa node (umumnya tiga salinan/replica) sehingga kegagalan satu node atau bahkan satu disk tidak menyebabkan kehilangan data, menjadikan Ceph pilihan shared storage paling andal untuk cluster HA produksi karena tidak memiliki single point of failure pada sisi storage-nya sendiri — berbeda dengan NFS yang bergantung pada satu server tunggal.

```
root@pve1:~# pveceph install
root@pve1:~# pveceph init --network 10.10.30.0/24
root@pve1:~# pveceph mon create
root@pve1:~# pveceph osd create /dev/sdb
root@pve1:~# pveceph pool create itnsa-pool --size 3 --min_size 2
```

Parameter size 3 menetapkan tiga salinan data pada tiga node berbeda, sedangkan min_size 2 menetapkan bahwa pool tetap dapat menerima operasi baca/tulis selama minimal dua dari tiga salinan tersedia — konfigurasi yang selaras dengan prinsip quorum pada bagian 11.5. Meski sangat andal, Ceph memerlukan sumber daya perangkat keras lebih tinggi (disk khusus, RAM tambahan, jaringan berkecepatan tinggi) dibandingkan NFS sederhana, sehingga pemilihannya perlu mempertimbangkan skala kebutuhan riil infrastruktur.

Sumber: Proxmox VE Administration Guide, bagian Ceph RADOS Block Device (RBD).

11.12 Storage Replication sebagai Alternatif Shared Storage

Untuk cluster skala kecil yang belum mampu menyediakan shared storage penuh (NFS/iSCSI/Ceph), Proxmox VE menyediakan fitur storage replication berbasis ZFS (pvesr) sebagai alternatif: disk VM disimpan secara lokal pada tiap node, namun secara berkala (misalnya setiap 15 menit) direplikasi secara asinkron ke node lain yang menjadi target HA-nya.

```
root@pve1:~# pvesr create-local-job 201-0 pve2 --schedule "*/15"
root@pve1:~# pvesr status
```

Perbedaan penting dibandingkan shared storage sesungguhnya adalah sifat asinkron replikasi ini: apabila node sumber gagal di antara dua jadwal replikasi, perubahan data pada rentang waktu tersebut (misalnya beberapa menit terakhir) berpotensi hilang saat VM dijalankan ulang dari salinan replika di node tujuan. Meski demikian, pendekatan ini tetap jauh lebih baik dibandingkan tanpa HA sama sekali, dan menjadi pilihan populer untuk homelab maupun laboratorium sekolah dengan keterbatasan infrastruktur storage bersama.

11.13 HA Simulator untuk Pembelajaran

Proxmox VE menyediakan HA Simulator, sebuah utilitas yang memungkinkan peserta didik mempelajari dan menguji seluruh perilaku HA Manager (start, stop, migrate, fencing) pada cluster tiga node virtual secara simulasi di satu mesin saja, tanpa memerlukan tiga unit perangkat keras maupun tiga VM Proxmox terpisah. Simulator ini sangat berguna sebagai sarana pembelajaran konsep HA sebelum peserta didik mempraktikkannya pada cluster nyata (bagian 11.4), karena memungkinkan eksperimen "mematikan" node secara aman berulang kali tanpa risiko kerusakan pada lingkungan produksi.

```
root@pve:~# ha-manager simulator start
```

Melalui simulator, peserta didik dapat mengamati transisi status service HA (started → fence → recovery → started kembali di node lain) secara bertahap dan lebih mudah dipahami dibandingkan langsung mengamatinya pada cluster fisik yang prosesnya berlangsung sangat cepat.

Sumber: Proxmox VE High Availability Documentation, bagian HA Simulator.

11.14 Perbandingan dengan Solusi HA Virtualisasi Lain

Agar peserta didik memiliki wawasan yang lebih luas mengenai ekosistem virtualisasi enterprise, berikut perbandingan singkat pendekatan HA pada Proxmox VE dengan dua solusi HA virtualisasi populer lainnya.

Aspek	Proxmox VE HA	VMware vSphere HA/DRS	Microsoft Hyper-V Failover Clustering
Lisensi	Open-source, gratis untuk fitur inti	Berbayar (lisensi vSphere + vCenter)	Termasuk dalam Windows Server Datacenter/Standard
Komponen cluster	Corosync + pmxcfs (terintegrasi)	vCenter Server (terpisah)	Failover Cluster Manager (peran Windows Server)
Mekanisme fencing	Watchdog (software/hardware)	Host isolation response berbasis heartbeat	Quorum witness (disk/share/cloud)
Live migration	vMotion-equivalent (qm migrate)	vMotion	Live Migration

Perbandingan ini menegaskan bahwa konsep inti HA — quorum, fencing, live migration, resource prioritization — bersifat universal di seluruh platform virtualisasi enterprise, meski istilah dan implementasi teknisnya berbeda. Kompetensi yang dipelajari peserta didik pada Proxmox VE dalam buku ini pada dasarnya dapat ditransfer dengan mudah ke platform lain yang mungkin dijumpai di dunia kerja.

11.15 Troubleshooting Umum pada Cluster HA

Gejala	Kemungkinan Penyebab	Langkah Penanganan
Node terus-menerus reboot sendiri (fencing loop)	Jaringan Corosync tidak stabil sehingga quorum terus hilang-muncul	Periksa latensi jaringan cluster, pertimbangkan NIC khusus untuk Corosync, cek dengan corosync-cfgtool -s
Cluster kehilangan quorum saat satu node dimatikan (cluster 2 node)	Jumlah node genap tanpa QDevice	Tambahkan QDevice atau node ketiga sesuai bagian 11.5
VM tidak bisa di-live migrate	VM menggunakan local storage, bukan shared storage	Pindahkan disk VM ke storage bersama (NFS/Ceph) sesuai bagian 10.14
VM tidak otomatis pindah saat node mati	VM belum didaftarkan sebagai resource HA	Tambahkan VM menggunakan ha-manager add sesuai bagian 11.7

11.16 Perencanaan Kapasitas Cluster: Prinsip N+1

Perencanaan kapasitas yang baik pada cluster HA mengikuti prinsip redundansi N+1, yaitu kapasitas total cluster dirancang agar tetap mampu menampung seluruh beban kerja (VM/container) meski salah satu node hilang dari layanan. Sebagai ilustrasi, apabila ITNSA.ID memiliki total kebutuhan 24 GB RAM untuk seluruh VM (int-srv, mail, web-01, web-02, gateway) pada cluster tiga node, maka setiap node idealnya dialokasikan

kapasitas yang cukup menampung sepertiga beban ditambah cadangan, sehingga ketika satu node gagal, dua node yang tersisa masih mampu menjalankan seluruh VM tanpa kehabisan sumber daya.

Skenario Kapasitas	Total RAM Cluster	Dampak saat 1 Node Gagal
Kapasitas pas-pasan (tanpa N+1)	3 node x 8 GB = 24 GB (sama dengan total kebutuhan)	VM gagal dijalankan ulang karena 2 node tersisa (16 GB) tidak cukup menampung 24 GB kebutuhan
Kapasitas dengan N+1	3 node x 12 GB = 36 GB (150% dari kebutuhan)	VM tetap dapat dijalankan ulang sepenuhnya pada 2 node tersisa (24 GB tersedia, cukup untuk 24 GB kebutuhan)

Prinsip ini menjelaskan mengapa cluster HA produksi jarang dijalankan pada kapasitas mendekati 100 persen di kondisi normal — sengaja disisakan headroom agar proses failover pada bagian 11.7–11.9 benar-benar dapat berjalan, bukan hanya terdaftar sebagai resource HA namun gagal dijalankan ulang karena kehabisan sumber daya di node yang tersisa.

11.17 Irisan dengan Mata Pelajaran Keamanan Jaringan

Sebagaimana ditegaskan pada Bab 1, buku ini berfokus pada aspek fungsional (membangun agar layanan berjalan). Cluster Proxmox VE beserta HA yang telah dibangun pada bab ini juga memiliki permukaan serangan (attack surface) tersendiri — misalnya web interface administrasi, komunikasi Corosync antar node, dan akses API — yang pengerasannya (hardening) merupakan ranah mata pelajaran Keamanan Jaringan. Tabel berikut menggambarkan pembagian tersebut secara konkret pada konteks Proxmox.

Topik	Dibahas di Administrasi Sistem Jaringan (bab ini)	Dibahas di Keamanan Jaringan
Web interface Proxmox	Cara mengakses dan menggunakan (Bab 10)	Pembatasan akses, autentikasi dua faktor, sertifikat TLS resmi
Jaringan Corosync	Cara mengonfigurasi agar cluster terbentuk (Bab 11)	Isolasi VLAN khusus, enkripsi, deteksi anomali trafik
Firewall Proxmox	Cara mengaktifkan dan membuat aturan dasar (Bab 10)	Perancangan kebijakan firewall berlapis dan audit berkala

11.18 Studi Kasus: Migrasi Infrastruktur ITNSA.ID ke Cluster HA

Sebagai rangkuman penerapan menyeluruh Bab 10 dan Bab 11, berikut tahapan migrasi infrastruktur ITNSA.ID dari VM mandiri (Bab 2–9, dijalankan pada VMware Workstation atau node Proxmox tunggal) menjadi cluster Proxmox VE dengan HA penuh, sebagaimana akan dipraktikkan pada proyek terintegrasi di Bab 12.

1. Audit kebutuhan sumber daya seluruh VM/container ITNSA.ID (CPU, RAM, disk) untuk menentukan spesifikasi minimal setiap node cluster mengikuti prinsip N+1 pada bagian 11.16.
2. Instalasi Proxmox VE pada tiga node fisik/virtual dengan hostname dan alamat IP final (tidak dapat diubah setelah cluster terbentuk).
3. Pembentukan cluster menggunakan pvecm create dan pvecm add sesuai bagian 11.4, diikuti verifikasi status quorum.

4. Penyediaan shared storage (NFS untuk laboratorium, atau Ceph untuk simulasi produksi) sesuai bagian 10.14 dan 11.11.
5. Migrasi VM/container yang sudah ada (int-srv, mail, web-01, web-02, gateway) ke shared storage tersebut menggunakan fitur move disk pada Proxmox.
6. Pendaftaran seluruh VM/container sebagai resource HA dengan HA Group dan prioritas sesuai rancangan pada bagian 11.10.
7. Pengujian menyeluruh: simulasi kegagalan setiap node secara bergantian, verifikasi seluruh layanan (DNS, LDAP, Mail, Web, Gateway, VPN) tetap dapat diakses client sesuai skenario pengujian end-to-end pada Bab 9.
8. Dokumentasi hasil migrasi, termasuk waktu downtime yang terukur pada setiap skenario kegagalan, sebagai bahan laporan proyek terintegrasi.

11.19 Mengukur Ketersediaan Layanan: Konsep Uptime dan SLA

Efektivitas suatu implementasi HA lazim diukur menggunakan persentase ketersediaan (uptime) dalam periode satu tahun, yang sering dituangkan sebagai Service Level Agreement (SLA) antara penyedia layanan IT dengan penggunanya. Semakin tinggi angka persentase yang dijanjikan, semakin sedikit total downtime yang diizinkan dalam setahun.

Tingkat Ketersediaan	Sebutan Umum	Total Downtime yang Diizinkan per Tahun
99%	Two nines	Sekitar 3 hari 15 jam
99,9%	Three nines	Sekitar 8 jam 45 menit
99,99%	Four nines	Sekitar 52 menit
99,999%	Five nines	Sekitar 5 menit

Mengacu pada karakteristik restart-based HA Proxmox VE yang telah dibahas pada bagian 11.9 (downtime 1–3 menit per kejadian kegagalan node), sebuah cluster ITNSA.ID dengan HA aktif secara realistis dapat menargetkan tingkat ketersediaan pada kisaran 99,9 persen (three nines) dengan asumsi kegagalan node tidak terjadi lebih dari beberapa kali dalam setahun. Untuk mendekati 99,99 persen atau lebih, diperlukan kombinasi tambahan seperti HA tingkat aplikasi (Bab 6) yang menyembunyikan sebagian besar downtime failover dari sudut pandang pengguna akhir, serta perangkat keras dengan hardware watchdog yang mempercepat proses fencing dibandingkan softdog.

11.20 Checklist Kesiapan Produksi Cluster HA

Sebagai penutup materi HA, berikut daftar periksa (checklist) yang dapat digunakan peserta didik maupun guru untuk mengevaluasi kesiapan sebuah cluster Proxmox VE sebelum dinyatakan siap menopang layanan produksi ITNSA.ID.

- Cluster terdiri dari minimal tiga node (atau dua node dengan QDevice aktif).
- Seluruh node menjalankan versi Proxmox VE yang identik dan telah diperbarui (apt update && apt dist-upgrade).
- Jaringan Corosync menggunakan NIC terdedikasi, terpisah dari trafik VM dan storage.
- Shared storage (NFS/iSCSI/Ceph) telah diuji dapat diakses oleh seluruh node secara bersamaan.
- Kapasitas cluster memenuhi prinsip N+1 sesuai perhitungan pada bagian 11.16.

- Seluruh VM kritikal (int-srv, mail, gateway) telah didaftarkan sebagai resource HA dengan HA Group dan prioritas yang sesuai.
- Mekanisme fencing telah diuji dan dipastikan berfungsi (baik softdog maupun hardware watchdog).
- Notifikasi email/webhook untuk peristiwa HA (fencing, failover) telah dikonfigurasi sesuai bagian 10.16.
- Backup rutin (vzdump/PBS) tetap dijalankan meski HA sudah aktif — HA melindungi dari kegagalan perangkat keras, bukan dari kesalahan konfigurasi maupun serangan siber yang dapat merusak data pada seluruh salinan sekaligus.
- Prosedur failover telah didokumentasikan dan diujicobakan secara berkala, bukan hanya dikonfigurasi lalu diasumsikan berfungsi tanpa pernah diuji.

11.21 Glosarium Istilah HA dan Cluster Proxmox VE

Berikut kumpulan istilah kunci yang telah muncul sepanjang Bab 11, disusun sebagai rujukan cepat bagi peserta didik maupun guru.

Istilah	Penjelasan Singkat
Corosync	Lapisan komunikasi cluster yang menjaga seluruh node saling bertukar heartbeat
pmxcfs	Sistem berkas cluster Proxmox yang mereplikasi konfigurasi ke seluruh node secara real-time
Quorum	Jumlah suara minimum (mayoritas) yang diperlukan cluster untuk mengambil keputusan secara aman
QDevice	Perangkat ringan penyedia satu suara tambahan sebagai tie-breaker pada cluster dengan jumlah node genap
Fencing	Mekanisme memastikan node yang gagal benar-benar berhenti sebelum VM-nya dipindahkan, mencegah split-brain
Watchdog	Timer perangkat keras/lunak yang memaksa reboot node apabila tidak disegarkan secara berkala oleh HA stack
Split-brain	Kondisi bahaya ketika VM yang sama berjalan bersamaan di dua node berbeda, berpotensi merusak data
CRM	Cluster Resource Manager, pengambil keputusan HA pada tingkat cluster
LRM	Local Resource Manager, pelaksana perintah CRM pada masing-masing node
HA Group	Kumpulan aturan yang menentukan node mana saja yang boleh menjalankan suatu VM beserta prioritasnya
Live Migration	Perpindahan VM antar node tanpa downtime, dilakukan secara terencana pada kondisi node sehat
Ceph	Sistem penyimpanan terdistribusi terintegrasi Proxmox yang mereplikasi data ke banyak node
N+1 Redundancy	Prinsip perencanaan kapasitas agar cluster tetap sanggup menampung seluruh beban meski satu node hilang

11.22 Rubrik Penilaian Praktik High Availability

Komponen Penilaian	Indikator	Bobot
Pembentukan cluster	Cluster tiga node berhasil terbentuk dengan status quorum OK	20%

Komponen Penilaian	Indikator	Bobot
Konfigurasi shared storage	Shared storage dapat diakses seluruh node dan VM berhasil dijalankan di atasnya	20%
Pendaftaran resource HA	VM terdaftar sebagai resource HA dengan status dan HA Group yang sesuai rancangan	20%
Pengujian failover	Simulasi kegagalan node berhasil memicu fencing dan VM berjalan ulang di node lain, waktu downtime tercatat	30%
Dokumentasi dan analisis	Laporan pengujian lengkap, termasuk perbandingan dengan HA tingkat aplikasi pada Bab 6	10%

11.23 Sumber Belajar Lanjutan

Bagi peserta didik yang ingin memperdalam materi HA Proxmox VE secara mandiri di luar cakupan buku ini, berikut sumber rujukan resmi dan komunitas yang direkomendasikan.

Sumber	Cakupan
pve.proxmox.com/pve-docs	Dokumentasi resmi lengkap, termasuk chapter-ha-manager dan chapter-pvecm yang menjadi rujukan utama bab ini
forum.proxmox.com	Forum komunitas resmi, tempat bertanya dan mempelajari studi kasus troubleshooting HA dari pengguna di seluruh dunia
pve.proxmox.com/wiki	Wiki komunitas berisi tutorial tambahan, termasuk konfigurasi hardware watchdog untuk berbagai merek server
Bugzilla dan mailing list Proxmox	Pelacakan isu teknis dan diskusi pengembangan fitur, berguna untuk memahami perkembangan HA Manager dari versi ke versi

11.10 Menerapkan Konsep HA pada Infrastruktur ITNSA.ID

Sebagai penerapan menyeluruh, seluruh VM/container yang membentuk infrastruktur ITNSA.ID pada Bab 2 hingga Bab 9 (int-srv, mail, web-01, web-02, gateway) dapat dijalankan di atas cluster Proxmox VE tiga node, dengan skema prioritas HA sebagai berikut.

VM/Container	Prioritas HA	Pertimbangan
int-srv (DNS, LDAP, CA)	Tinggi	Seluruh layanan lain bergantung pada int-srv; downtime di sini berdampak luas
mail	Tinggi	Layanan bisnis kritikal yang harus segera pulih bila node gagal
web-01, web-02	Sedang	Sudah memiliki HA tingkat aplikasi (Bab 6) sehingga kegagalan satu VM tidak langsung mematikan layanan web secara keseluruhan
gateway	Tinggi	Kegagalan gateway memutus seluruh akses internal maupun akses dari luar

Pengaturan ini menunjukkan bahwa HA tingkat infrastruktur (Bab 11) dan HA tingkat aplikasi (Bab 6) saling melengkapi: web-01 dan web-02 tetap didaftarkan sebagai resource HA agar VM-nya otomatis pulih bila

node gagal, namun karena keduanya sudah memiliki failover tingkat aplikasi melalui Keepalived dan HAProxy, dampak downtime saat proses relocation berlangsung dapat diminimalkan lebih jauh.

11.10.1 Ilustrasi Dampak: Sebelum dan Sesudah Penerapan HA

Untuk menegaskan manfaat konkret dari seluruh konfigurasi pada bab ini, berikut ilustrasi perbandingan dampak kegagalan node fisik terhadap layanan ITNSA.ID sebelum dan sesudah cluster HA diterapkan.

Skenario	Tanpa Cluster HA (Bab 10 saja)	Dengan Cluster HA (Bab 10 + Bab 11)
Node fisik tempat int-srv berjalan mati mendadak	DNS, LDAP, dan CA lumpuh total hingga admin datang dan memulihkan VM secara manual (bisa berjam-jam, tergantung ketersediaan admin)	VM int-srv otomatis di-fencing dan dijalankan ulang di node lain dalam 1–3 menit tanpa campur tangan manual
Node fisik tempat mail berjalan mengalami kerusakan hard disk	Layanan email terhenti hingga hard disk diganti dan VM dipulihkan dari backup terakhir (berpotensi kehilangan data sejak backup terakhir)	VM mail berjalan dari shared storage yang tidak terpengaruh kerusakan disk lokal node, langsung pulih di node lain
Maintenance terjadwal (pembaruan firmware node)	Seluruh VM pada node tersebut wajib dimatikan selama maintenance, downtime penuh sesuai durasi maintenance	VM dipindahkan lebih dulu menggunakan live migration (bagian 11.9) tanpa downtime, node dapat di-maintenance dengan aman

Ilustrasi ini menegaskan bahwa investasi waktu mempelajari dan mengonfigurasi cluster HA pada bab ini secara langsung diterjemahkan menjadi pengurangan risiko downtime yang signifikan bagi kelangsungan bisnis ITNSA.ID, sekaligus memberikan fleksibilitas operasional (seperti maintenance tanpa downtime) yang tidak mungkin dicapai hanya dengan node Proxmox tunggal seperti pada Bab 10.

🔧 Praktik: Membangun Cluster HA Proxmox VE untuk Infrastruktur ITNSA.ID

1. Siapkan tiga node Proxmox VE (dapat berupa nested VM pada VirtualBox jika keterbatasan perangkat keras), pastikan seluruh node menggunakan versi yang sama dan saling terhubung dalam satu jaringan.

2. Bentuk cluster menggunakan pvecm create pada node pertama, lalu gabungkan dua node lain menggunakan pvecm add.

3. Siapkan shared storage sederhana menggunakan NFS server (dapat memanfaatkan salah satu container Linux dari Bab 10) dan pasang (mount) sebagai storage bersama pada ketiga node.

4. Buat satu VM uji coba pada shared storage tersebut, daftarkan sebagai resource HA menggunakan ha-manager add, lalu verifikasi status menggunakan ha-manager status.

5. Lakukan simulasi kegagalan node dengan mematikan paksa (power off) node tempat VM tersebut berjalan, amati proses fencing pada log (journalctl -u pve-ha-crm -f) dan catat waktu yang dibutuhkan hingga VM kembali berjalan di node lain.

6. Buat HA Group dengan prioritas tertentu, kaitkan VM uji coba ke grup tersebut, lalu uji apakah VM kembali (fail back) ke node prioritas utama setelah node tersebut dinyalakan kembali.

7. Bandingkan waktu downtime yang tercatat pada simulasi ini dengan waktu downtime pengujian failover HAProxy/Keepalived pada Bab 6, lalu diskusikan perbedaan karakteristik kedua pendekatan HA tersebut.

Contoh dalam Kehidupan Nyata: HA Proxmox di balik layanan cloud dan hosting Indonesia

> *Banyak penyedia layanan cloud dan hosting lokal di Indonesia membangun infrastrukturnya di atas cluster Proxmox VE dengan konfigurasi HA serupa yang dipelajari pada bab ini, karena kombinasi biaya lisensi nol*

dan keandalan tingkat enterprise sangat menguntungkan secara bisnis. Ketika pelanggan hosting mengalami downtime singkat pada VPS-nya yang diikuti pesan "server sedang dipulihkan otomatis", kemungkinan besar yang terjadi di balik layar adalah proses relocation HA seperti yang baru saja dipraktikkan: node fisik tempat VPS tersebut berjalan mengalami gangguan, watchdog memicu fencing, dan CRM cluster memerintahkan VPS tersebut dijalankan ulang di node fisik lain yang sehat — seluruhnya berlangsung otomatis tanpa campur tangan manual dalam hitungan menit, jauh lebih cepat dibandingkan jika teknisi harus memperbaiki perangkat keras yang rusak terlebih dahulu.

Aktivitas Pembelajaran

1. Praktik membangun cluster tiga node Proxmox VE beserta verifikasi status quorum menggunakan pvecm status.

2. Praktik mendaftarkan VM sebagai resource HA dan menguji mekanisme failover melalui simulasi kegagalan node.

3. Praktik membuat HA Group dengan prioritas tertentu dan mengamati perilaku fail back.

4. Diskusi kelompok: mengapa cluster HA minimal memerlukan tiga node (atau dua node dengan QDevice), dan apa yang terjadi jika syarat ini diabaikan?

5. Proyek mini: peserta didik menyusun rancangan penempatan seluruh VM/container ITNSA.ID pada cluster tiga node beserta skema prioritas HA-nya, dipresentasikan di depan kelas.

Rangkuman

High Availability tingkat infrastruktur pada Proxmox VE dibangun di atas cluster minimal tiga node yang berkomunikasi melalui Corosync dan pmxcfs, dengan mekanisme quorum untuk mencegah keputusan yang tidak aman saat sebagian node terputus. Fencing berbasis watchdog memastikan tidak terjadi split-brain dengan cara memaksa node yang kehilangan quorum melakukan self-reset sebelum VM-nya dijalankan ulang di node lain melalui HA Manager (CRM dan LRM). HA Proxmox bersifat restart-based dengan downtime singkat (bukan zero-downtime), sehingga idealnya dikombinasikan dengan HA tingkat aplikasi seperti Keepalived dan HAProxy pada Bab 6 untuk keandalan maksimal pada infrastruktur ITNSA.ID.

Soal Latihan / Refleksi

1. Jelaskan perbedaan antara High Availability tingkat infrastruktur (Bab 11) dan HA tingkat aplikasi (Bab 6), serta bagaimana keduanya saling melengkapi.

2. Jelaskan mengapa cluster Proxmox VE minimal memerlukan tiga node untuk quorum yang andal, dan bagaimana QDevice dapat menjadi solusi pada cluster dua node.

3. Jelaskan mekanisme fencing berbasis watchdog pada Proxmox VE dan mengapa mekanisme ini penting untuk mencegah split-brain.

4. Jelaskan perbedaan live migration dan relocation (failover HA), termasuk dampaknya terhadap downtime layanan.

5. Rancang skema HA Group untuk lima VM/container infrastruktur ITNSA.ID pada cluster tiga node, lengkap dengan alasan penentuan prioritasnya.

6. Jelaskan prinsip N+1 dalam perencanaan kapasitas cluster HA beserta konsekuensinya apabila prinsip ini diabaikan.

7. Jelaskan mengapa backup rutin tetap wajib dijalankan meski cluster HA sudah aktif, padahal HA sudah melindungi dari kegagalan node.

BAB 12 — Proyek Terintegrasi dan Evaluasi Akhir

Tujuan Pembelajaran

- Peserta didik mampu mengintegrasikan seluruh layanan yang dipelajari menjadi satu infrastruktur utuh dalam sebuah proyek simulasi topologi.
- Peserta didik mampu mengerjakan evaluasi akhir sebagai bentuk asesmen sumatif materi Administrasi Sistem Jaringan.

12.1 Keterkaitan Antar Bab

Sebagaimana ditegaskan sejak Bab 1, seluruh bab dalam buku ini bukan merupakan topik yang berdiri sendiri, melainkan bagian dari satu infrastruktur perusahaan ITNSA.ID yang saling terhubung. DNS (Bab 2) menjadi fondasi resolusi nama bagi seluruh layanan lain. LDAP (Bab 3) menjadi basis identitas pengguna yang dipakai autentikasi Mail Server (Bab 5). PKI/CA (Bab 4) menerbitkan sertifikat yang dipakai mengenkripsi komunikasi Mail Server (Bab 5) maupun Web Server (Bab 6). Ansible (Bab 7) mengotomasi konfigurasi yang sebelumnya dilakukan manual pada Bab 2 hingga Bab 6. Gateway (Bab 8) menjembatani seluruh layanan DMZ dengan dunia luar melalui NAT, port forwarding, dan VPN. Bab 9 menjadi tahap pembuktian akhir bahwa seluruh rangkaian tersebut benar-benar berfungsi dari sudut pandang pengguna. Bab 10 dan Bab 11 melengkapi seluruh rangkaian tersebut dari sisi platform: seluruh VM/container ITNSA.ID pada dasarnya dapat dijalankan di atas Proxmox VE (Bab 10), dan agar layanan tetap tersedia meski terjadi kegagalan perangkat keras server fisik, seluruh VM tersebut dikelola melalui cluster High Availability Proxmox (Bab 11) yang bekerja pada lapisan berbeda namun saling melengkapi dengan HA tingkat aplikasi pada Bab 6.

No	Elemen	Inti Materi
1	Administrasi DNS Server	BIND9, forward/reverse zone, record A/CNAME/MX/PTR
2	Administrasi Direktori Terpusat (LDAP)	slapd, struktur DN, LDIF, manajemen akun
3	Administrasi PKI/CA	OpenSSL, Root CA, penerbitan & pengelolaan sertifikat
4	Administrasi Mail Server	Postfix (SMTP), Dovecot (IMAP), TLS, integrasi LDAP
5	Administrasi Web Server & HA Aplikasi	Nginx, Keepalived (VRRP), HAProxy (load balancing & TLS termination)
6	Otomasi Administrasi Server	Ansible control node, playbook idempoten
7	Administrasi Gateway	NAT masquerade, port forwarding, WireGuard VPN
8	Pengujian Sisi Klien	Akun lokal, koneksi VPN, uji web/SSL, uji mail client
9	Virtualisasi Server (Proxmox VE)	KVM, LXC, manajemen user/permission, backup/restore, firewall, Fail2ban
10	High Availability Virtualization	Cluster Proxmox, quorum, fencing, HA Manager, live migration

12.2 Proyek Terintegrasi: Membangun Infrastruktur ITNSA.ID Utuh

Sebagai penutup rangkaian pembelajaran, disarankan peserta didik (secara berkelompok) membangun seluruh infrastruktur DNS-LDAP-CA-Mail-Web HA-Firewall-VPN-Client dalam satu simulasi topologi utuh

menggunakan VMware Workstation, sebagaimana skenario lengkap pada soal LKS. Proyek ini menjadi wahana asesmen kinerja (performance assessment) yang menilai kemampuan peserta didik mengintegrasikan seluruh kompetensi, bukan hanya menguasai satu layanan secara terpisah.

Tahapan Proyek	Deliverable yang Diharapkan
Perencanaan Topologi	Diagram jaringan lengkap (int-srv, mail, web-01, web-02, gateway, client) beserta skema pengalamatan IP setiap segmen
Implementasi Fondasi Internal	DNS, LDAP, dan Root CA berjalan dan dapat diverifikasi (dig, ldapsearch, openssl x509)
Implementasi Layanan Aplikasi	Mail server dan web server (dengan HA) berjalan, terintegrasi dengan DNS, LDAP, dan sertifikat
Otomasi	Minimal satu playbook Ansible yang berhasil diterapkan ke lebih dari satu server
Implementasi Gateway	NAT, port forwarding, dan VPN berfungsi, dapat diverifikasi dari client eksternal
Verifikasi Akhir	Laporan pengujian end-to-end dari sisi client, mencakup seluruh skenario pada Bab 9

Contoh dalam Kehidupan Nyata: Proyek terintegrasi sebagai simulasi dunia kerja nyata

> *Proyek pada bab ini secara sengaja dirancang meniru pekerjaan nyata seorang junior network administrator yang baru bergabung di sebuah perusahaan dan diminta membangun infrastruktur IT dari nol untuk kantor cabang baru. Dalam skenario nyata, tidak ada yang meminta hanya "instal DNS saja" atau "instal mail server saja" — pekerjaan sesungguhnya selalu berupa integrasi banyak layanan yang saling bergantung, persis seperti proyek terintegrasi ini. Kemampuan menyusun dokumentasi topologi dan laporan pengujian end-to-end juga mencerminkan praktik profesional dunia kerja, di mana setiap pekerjaan instalasi infrastruktur wajib didokumentasikan agar dapat diaudit, dipelihara, dan dipahami oleh anggota tim lain di kemudian hari.*

12.3 Soal Evaluasi Akhir

Soal berikut dapat digunakan guru sebagai bahan asesmen sumatif atau latihan persiapan kompetisi keahlian, mencakup seluruh bab dalam buku ini.

1. Jelaskan alur kerja resolusi nama domain mail.itnsa.id sejak diketik client hingga menghasilkan alamat IP, serta jenis record DNS yang berperan di dalamnya.

2. Jelaskan langkah membuat organizational unit dan akun pengguna baru pada LDAP, termasuk objectClass yang diperlukan agar akun dapat dipakai login sistem.

3. Jelaskan urutan proses membangun Root CA internal hingga menerbitkan sertifikat untuk sebuah server, beserta atribut Basic Constraints yang wajib disertakan pada sertifikat CA.

4. Jelaskan bagaimana Mail Server ITNSA.ID pada Bab 5 memanfaatkan hasil kerja Bab 2, 3, dan 4 secara terintegrasi.

5. Jelaskan mekanisme kerja Keepalived dan HAProxy dalam memastikan layanan web tetap tersedia meski satu server backend mati.

6. Jelaskan konsep idempoten pada Ansible beserta contoh penerapannya pada playbook manajemen pengguna.

7. Jelaskan perbedaan NAT masquerade dan port forwarding beserta contoh penerapannya pada gateway ITNSA.ID.

8. Jelaskan langkah mengonfigurasi client agar dapat terhubung ke VPN WireGuard secara otomatis saat boot serta menggunakan DNS internal perusahaan.

9. Jelaskan mengapa sertifikat Root CA internal perlu didistribusikan secara manual ke setiap client, berbeda dengan sertifikat dari CA publik.

10. Jelaskan perbedaan model virtualisasi KVM dan LXC pada Proxmox VE, beserta pertimbangan pemilihan masing-masing untuk layanan ITNSA.ID.

11. Jelaskan bagaimana High Availability tingkat infrastruktur (cluster Proxmox) dan HA tingkat aplikasi (Keepalived/HAProxy) saling melengkapi pada infrastruktur ITNSA.ID.

12. Rancang skenario pengujian end-to-end sederhana yang membuktikan seluruh infrastruktur ITNSA.ID (DNS, LDAP, PKI, Mail, Web HA, Gateway, VPN, dan Virtualisasi HA) berfungsi dengan baik dari sudut pandang seorang karyawan baru yang bekerja dari rumah.

12.4 Daftar Pustaka

Materi dalam buku ini disusun dan dikembangkan mengacu pada dokumen Target Pembelajaran Administrasi Sistem Jaringan Kelas XI Program Keahlian TJKT, yang bersumber dari Naskah Soal Lomba Kompetensi Siswa (LKS) SMK Tingkat Kabupaten Tasikmalaya Tahun 2026, Bidang IT Network System Administration, Modul A: Linux Environment. Materi Bab 10 dan Bab 11 diperkaya dari I Putu Hariyadi, Panduan Praktikum Administrasi Sistem Jaringan – Proxmox VE 8.0 (Universitas Bumigora, 2023), serta dokumentasi resmi Proxmox VE (pve.proxmox.com/pve-docs), khususnya bagian Cluster Manager dan High Availability. Guru disarankan melengkapi referensi dengan dokumentasi resmi masing-masing perangkat lunak yang digunakan, yaitu dokumentasi BIND9 dan ISC (Internet Systems Consortium), dokumentasi OpenLDAP, dokumentasi OpenSSL, dokumentasi Postfix dan Dovecot, dokumentasi Nginx/Keepalived/HAProxy, dokumentasi Ansible (docs.ansible.com), dokumentasi WireGuard (wireguard.com), serta dokumentasi Proxmox VE (pve.proxmox.com), untuk memastikan materi tetap relevan dengan perkembangan versi terbaru masing-masing perangkat lunak.

LAMPIRAN 1 — Referensi Cepat Perintah CLI Proxmox VE

Lampiran ini merangkum perintah command line interface (CLI) Proxmox VE yang paling sering digunakan sepanjang Bab 10 dan Bab 11, dikelompokkan berdasarkan kategori fungsinya, sebagai lembar rujukan cepat (cheat sheet) bagi peserta didik maupun guru saat praktik di laboratorium.

L.1 Manajemen VM (qm) dan Container (pct)

Perintah	Fungsi
qm create <vmid> [opsi]	Membuat VM KVM baru
qm start / stop / shutdown <vmid>	Menjalankan, mematikan paksa, atau mematikan normal VM
qm migrate <vmid> <node> --online	Melakukan live migration VM ke node lain
qm snapshot <vmid> <nama>	Membuat snapshot VM
qm rollback <vmid> <nama>	Mengembalikan VM ke kondisi snapshot
qm template <vmid>	Menjadikan VM sebagai template untuk clone
qm clone <vmid> <newid>	Menggandakan VM dari template
pct create <vmid> <template> [opsi]	Membuat LXC container baru dari template

Perintah	Fungsi
pct start / stop / enter <vmid>	Menjalankan, mematikan, atau masuk ke console container

L.2 Manajemen Cluster dan HA (pvecm, ha-manager)

Perintah	Fungsi
pvecm create <nama-cluster>	Membentuk cluster baru pada node pertama
pvecm add <ip-node-pertama>	Menggabungkan node ke cluster yang sudah ada
pvecm status	Menampilkan status quorum dan keanggotaan cluster
pvecm nodes	Menampilkan daftar node anggota cluster
pvecm qdevice setup <ip-qdevice>	Menambahkan QDevice sebagai suara tambahan
ha-manager add vm:<vmid> --state started	Mendaftarkan VM sebagai resource HA
ha-manager status	Menampilkan status HA Manager (quorum, master, service)
ha-manager groupadd <nama> --nodes <node:priority,...>	Membuat HA Group dengan prioritas node
ha-manager set vm:<vmid> --group <nama>	Mengaitkan VM ke suatu HA Group

L.3 Manajemen Storage dan Backup

Perintah	Fungsi
pvesm status	Menampilkan status seluruh storage yang terdaftar
pveceph install / init / osd create	Menginstal dan mengonfigurasi Ceph sebagai shared storage
pvesr create-local-job <vmid>-<disk> <node>	Membuat jadwal storage replication (ZFS)
vzdump <vmid> --mode snapshot	Membuat backup VM/container dalam kondisi berjalan
qmrestore <file-backup> <vmid-baru>	Memulihkan VM dari berkas backup

L.4 Manajemen User, Repository, dan Keamanan Dasar

Perintah	Fungsi
pveum user add <user@realm>	Membuat akun pengguna baru pada Proxmox VE
pveum aclmod <path> --user <user> --role <role>	Memberikan hak akses (role) pada path tertentu
pveum user list	Menampilkan daftar seluruh pengguna
apt update / apt dist-upgrade	Memperbarui daftar paket dan sistem sesuai repository aktif
systemctl status/restart fail2ban	Memeriksa/me-restart layanan proteksi brute force

Peserta didik disarankan mencetak atau menyimpan lampiran ini sebagai lembar rujukan pribadi selama sesi praktik di laboratorium, mengingat padatnya jumlah perintah CLI yang perlu dihafal maupun dipahami konteks penggunaannya pada Bab 10 dan Bab 11.

LAMPIRAN 2 — Rancangan Alokasi Alamat IP Infrastruktur ITNSA.ID

Lampiran ini merangkum seluruh alokasi alamat IP yang telah dibangun sepanjang Bab 1 hingga Bab 11, disusun sebagai satu tabel referensi terpadu yang memudahkan peserta didik memahami keterkaitan antar layanan saat mengerjakan proyek terintegrasi pada Bab 12.

Segmen/Perangkat	Alamat IP Contoh	Layanan yang Berjalan	Dibahas pada
int-srv	10.10.12.2/24	DNS (BIND9), LDAP (slapd), Root CA (OpenSSL)	Bab 2, 3, 4
mail	10.10.20.5/24	Postfix (SMTP), Dovecot (IMAP)	Bab 5
web-01	10.10.20.11/24	Nginx, Keepalived (MASTER)	Bab 6
web-02	10.10.20.12/24	Nginx, Keepalived (BACKUP)	Bab 6
Virtual IP HAProxy	10.10.20.100/24	Load balancing & TLS termination	Bab 6
gateway (LAN)	10.10.12.1/24 & 10.10.20.1/24	NAT masquerade, port forwarding	Bab 8
gateway (WAN)	IP publik dinamis/statis	Pintu keluar internet, endpoint WireGuard	Bab 8
WireGuard tunnel	10.99.0.1/24 (gateway), 10.99.0.2/24 (client)	VPN client-to-site	Bab 8, 9
Ansible control node	10.10.12.2 (di int-srv)	Otomasi konfigurasi seluruh server	Bab 7
pve1 / pve2 / pve3	192.168.169.1–3/24 (lab nested)	Node cluster Proxmox VE	Bab 10, 11
QDevice (opsional)	192.168.169.10/24	Suara tambahan quorum cluster 2 node	Bab 11
NFS shared storage	192.168.169.20/24	Shared storage untuk live migration & HA	Bab 11

Skema alamat di atas bersifat contoh dan dapat disesuaikan dengan kondisi laboratorium masing-masing sekolah, namun konsistensi antar bab perlu dijaga agar peserta didik tidak kebingungan saat menghubungkan hasil kerja satu bab dengan bab lainnya pada proyek terintegrasi.

LAMPIRAN 3 — Daftar Singkatan

Singkatan	Kepanjangan
DNS	Domain Name System
LDAP	Lightweight Directory Access Protocol
PKI	Public Key Infrastructure
CA	Certificate Authority
CSR	Certificate Signing Request
TLS	Transport Layer Security
SMTP	Simple Mail Transfer Protocol
IMAP	Internet Message Access Protocol
MTA	Mail Transfer Agent
VRRP	Virtual Router Redundancy Protocol
VIP	Virtual IP
HA	High Availability
NAT	Network Address Translation
DNAT / SNAT	Destination/Source Network Address Translation
VPN	Virtual Private Network
IaC	Infrastructure as Code
KVM	Kernel-based Virtual Machine
LXC	Linux Container
CRM	Cluster Resource Manager (konteks HA Proxmox)
LRM	Local Resource Manager (konteks HA Proxmox)
PBS	Proxmox Backup Server
SLA	Service Level Agreement
ITNSA.ID	Nama perusahaan studi kasus (International Technology Network Solutions & Administration)

LAMPIRAN 4 — Lembar Kerja Siswa: Proyek Integrasi Infrastruktur ITNSA.ID

Lembar kerja ini digunakan sebagai instrumen pemantauan kemajuan peserta didik/kelompok dalam mengerjakan proyek terintegrasi pada Bab 12, mencakup seluruh capaian dari Bab 2 hingga Bab 11. Guru dapat memperbanyak lembar ini sebagai formulir cetak maupun digital (misalnya Google Form) untuk setiap kelompok.

Nama Kelompok	……
Kelas	……
Tanggal Mulai	……
Tanggal Selesai	……

No	Komponen Infrastruktur	Kriteria Selesai (Definition of Done)
1	DNS Server (BIND9)	Forward & reverse zone aktif; dig berhasil resolve
2	LDAP (slapd)	Minimal 3 akun pengguna berhasil dibuat & ldapsearch berhasil
3	PKI/CA (OpenSSL)	Root CA & sertifikat web/mail terbit; openssl x509 -text berhasil
4	Mail Server (Postfix & Dovecot)	Email terkirim & diterima; TLS aktif; login LDAP berhasil
5	Web Server & HA Aplikasi	Failover Keepalived teruji; load balancing HAProxy teruji
6	Otomasi (Ansible)	Playbook user & deployment berjalan idempoten
7	Gateway (NAT, Port Forward, VPN)	Client internal akses internet; WireGuard client tersambung
8	Pengujian Sisi Klien	VPN otomatis saat boot; SSL diterima; mail client berfungsi
9	Virtualisasi Proxmox VE	VM KVM & LXC berjalan; backup/restore & Fail2ban teruji
10	HA Cluster Proxmox	Cluster quorum OK; failover VM teruji; waktu downtime tercatat

Aspek Refleksi Akhir Kelompok	Catatan
Kendala teknis terbesar yang dihadapi	
Bagian yang paling mudah dipahami	
Bagian yang memerlukan bimbingan tambahan	
Rencana perbaikan bila proyek diulang	

Paraf Peserta Didik	Paraf Guru Pembimbing
……	……

LAMPIRAN 5 — Rencana Pembelajaran Bab 10 dan Bab 11 per Pertemuan

Mengingat padatnya materi Bab 10 dan Bab 11, lampiran ini menyarankan pembagian alokasi waktu menjadi 13 kali pertemuan (disesuaikan dengan jumlah jam pelajaran per minggu di masing-masing sekolah), agar guru dapat merencanakan Rencana Pelaksanaan Pembelajaran (RPP) secara bertahap tanpa terburu-buru menuntaskan seluruh materi virtualisasi dan HA dalam waktu singkat.

Pertemuan	Materi	Sub-bagian	Fokus Praktik
1	Konsep virtualisasi, hypervisor, instalasi Proxmox VE	10.1 – 10.4	Instalasi Proxmox sebagai nested VM
2	Subscription, repository, arsitektur Proxmox	10.5 – 10.7	Konfigurasi pasca-instalasi
3	VM KVM: MikroTik CHR sebagai gateway	10.8	Membuat & mengonfigurasi VM CHR
4	LXC container, user/permission	10.9 – 10.10	Membuat container CentOS & akun terbatas
5	Backup/restore, firewall, Fail2ban	10.11 – 10.12	Uji backup-restore & proteksi brute force
6	Jaringan virtual, storage, PBS, monitoring	10.13 – 10.16	Konfigurasi VLAN-aware bridge
7	Template, cloud-init, integrasi Ansible, TCO	10.17 – 10.25	Clone VM dari template
8	Konsep HA infrastruktur vs aplikasi, arsitektur cluster	11.1 – 11.3	Diskusi & perencanaan cluster 3 node
9	Membangun cluster, quorum, QDevice	11.4 – 11.5	Praktik pvecm create/add
10	Fencing, watchdog, HA Manager	11.6 – 11.7	Simulasi kegagalan node
11	HA Group, live migration, studi kasus ITNSA.ID	11.8 – 11.10	Uji failover & fail back
12	Ceph, storage replication, HA simulator	11.11 – 11.13	Eksplorasi shared storage lanjutan
13	Kapasitas, checklist produksi, evaluasi	11.14 – 11.22	Presentasi rancangan cluster HA ITNSA.ID

Guru dapat menyesuaikan urutan dan kedalaman materi sesuai ketersediaan perangkat keras laboratorium — misalnya bila hanya tersedia satu unit komputer per kelompok, pertemuan 8–13 (praktik cluster tiga node) dapat dilakukan secara bergiliran antar kelompok atau disederhanakan menjadi simulasi menggunakan HA Simulator (bagian 11.13) sebagai alternatif ketika perangkat keras fisik terbatas.

LAMPIRAN 6 — Contoh Soal Praktik Bergaya Kompetisi Keahlian (Proxmox & HA)

Lampiran ini menyajikan satu contoh skenario soal praktik komprehensif bergaya Lomba Kompetensi Siswa (LKS), yang dapat digunakan guru sebagai bahan ujian praktik akhir Bab 10 dan Bab 11 maupun simulasi persiapan kompetisi keahlian.

Skenario

Perusahaan ITNSA.ID berencana memindahkan seluruh infrastruktur server internal (int-srv, mail, web-01, web-02, gateway) yang sebelumnya berjalan pada VM mandiri, ke sebuah cluster Proxmox VE beranggotakan tiga node dengan High Availability aktif. Sebagai calon teknisi jaringan yang direkrut ITNSA.ID, Anda diminta menyelesaikan tugas berikut dalam waktu 180 menit.

Tugas Praktik

1. Instalasi: Instalasikan Proxmox VE 8.0 pada tiga node yang telah disediakan (dapat berupa nested VM), lalu nonaktifkan notifikasi "No valid subscription" dan alihkan repository ke jalur no-subscription.

2. Cluster: Bentuk cluster bernama itnsa-cluster menggunakan ketiga node tersebut, verifikasi status quorum OK menggunakan pvecm status.

3. Storage: Siapkan shared storage NFS yang dapat diakses ketiga node, daftarkan sebagai storage bersama pada datacenter.

4. Virtualisasi: Buat satu VM KVM MikroTik CHR (sesuai spesifikasi minimum pada bagian 10.8) di atas shared storage tersebut, konfigurasikan sebagai gateway sederhana dengan NAT aktif.

5. Container: Buat satu LXC container CentOS 9 Stream di atas shared storage yang sama, instal dan aktifkan layanan OpenSSH.

6. HA: Daftarkan kedua VM/container tersebut sebagai resource HA dengan max_restart 2 dan max_relocate 1, kelompokkan dalam satu HA Group dengan prioritas node pertama sebagai node utama.

7. Pengujian Failover: Simulasikan kegagalan node utama (matikan paksa), catat waktu yang dibutuhkan hingga kedua resource kembali berjalan di node lain menggunakan bukti tangkapan layar ha-manager status sebelum dan sesudah kegagalan.

8. Keamanan Dasar: Aktifkan firewall Proxmox pada level datacenter dan pasang Fail2ban untuk melindungi web interface dari percobaan login paksa.

9. Dokumentasi: Susun laporan singkat (maksimal 2 halaman) berisi topologi, langkah kerja, dan hasil pengujian failover, dilengkapi kesimpulan mengenai kesiapan produksi cluster sesuai checklist pada bagian 11.20.

Kriteria Penilaian

Komponen	Bobot
Instalasi & konfigurasi dasar (langkah 1)	10%
Pembentukan cluster & shared storage (langkah 2–3)	20%
Virtualisasi VM dan container (langkah 4–5)	20%
Konfigurasi dan pengujian HA (langkah 6–7)	30%
Keamanan dasar (langkah 8)	10%

Komponen	Bobot
Dokumentasi dan analisis (langkah 9)	10%

Guru disarankan menyesuaikan durasi dan bobot penilaian sesuai kondisi laboratorium serta tingkat kemahiran peserta didik, dan dapat menambahkan variasi skenario (misalnya kegagalan dua node sekaligus, atau kehilangan koneksi jaringan Corosync) untuk peserta didik yang telah menguasai skenario dasar.