MATA KULIAH ADMINISTRASI SISTEM · CEN3303

Manajemen Paket Perangkat Lunak dan Pembaruan Sistem (Patching)

Pertemuan 6 · Sub-CPMK 2.4

Program Studi D3 Teknik Komputer, Politeknik Negeri Padang

Semester 3 · 3 SKS (2 Teori – 1 Praktik)

PETA PEMBELAJARAN

Agenda Pertemuan Hari Ini

01

Konsep Package Manager

APT, YUM/DNF, dan Windows Update pada Windows Server

02

Dependency & Repository

Keterkaitan antar paket dan sumber paket terpercaya

03

Strategi Patching Aman

Jenis pembaruan dan praktik aman untuk server produksi

04

Studi Kasus & Diskusi Kelompok

Kegagalan layanan pasca-pembaruan sistem

SUB-CPMK 2.4

Capaian Pembelajaran Pertemuan Ini

Sub-CPMK 2.4 Mahasiswa mampu mengelola paket perangkat lunak dan melakukan pembaruan sistem (patching) secara aman.



Memahami konsep

package manager dan repository.



Menjelaskan konsep dependency

antar paket perangkat lunak.



Merancang strategi patching

yang aman untuk server produksi.

Metode: Ceramah, Diskusi & Praktikum Lab · CPL Terkait: CPL-2, CPL-8

Menjaga Perangkat Lunak Tetap Mutakhir

Setelah layanan berjalan baik melalui pengelolaan proses dan systemd (Bab 5), administrator perlu memastikan perangkat lunak yang terpasang tetap mutakhir dan bebas dari celah keamanan yang telah diketahui.

Bab ini membahas cara kerja manajer paket serta strategi pembaruan sistem yang aman.



Dua risiko yang berlawanan. Banyak organisasi menjadi korban serangan siber karena menunda patch keamanan berbulan-bulan; organisasi lain justru mengalami gangguan layanan karena memperbarui tanpa pengujian memadai. Bab ini mengajarkan keseimbangan keduanya.

KAITAN ANTAR BAB



Bab 5 — Proses & Layanan

Layanan sudah berjalan dan dikendalikan systemd.



Bab 6 — Paket & Patching

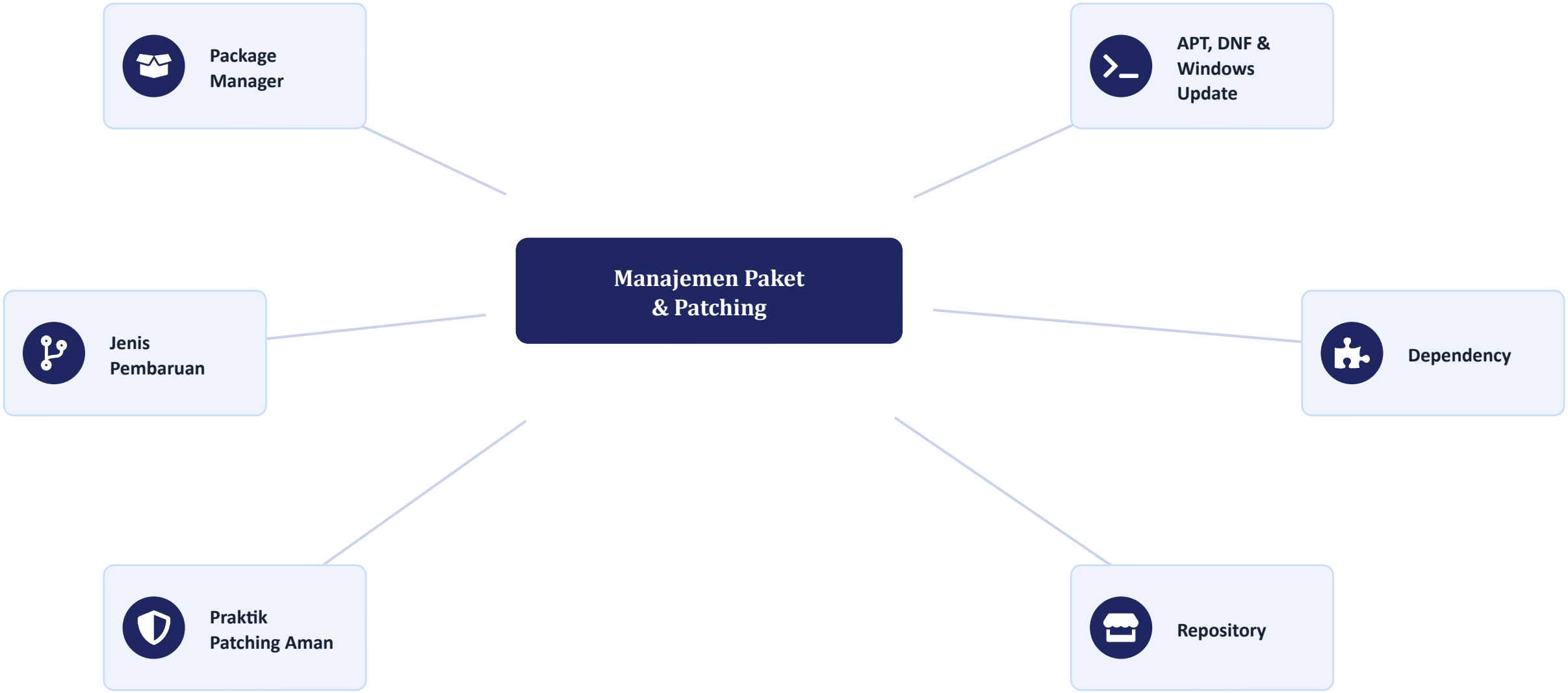
Menjaga perangkat lunak mutakhir dan aman (bab ini).



Bab 7 — Layanan DNS Server

Instalasi layanan jaringan bergantung pada manajemen paket.

Peta Konsep Bab 6



Konsep Package Manager

Package manager adalah alat yang mengotomatisasi proses instalasi, pembaruan, konfigurasi, dan penghapusan perangkat lunak — sehingga administrator tidak perlu mengunduh dan mengompilasi kode sumber secara manual.

Analogi Marketplace Ponsel — pengguna cukup menekan tombol "instal" tanpa perlu memahami detail proses pengunduhan, pemasangan dependensi, dan konfigurasi di baliknya.

SIKLUS YANG DIOTOMATISASI

Instalasi

Mengunduh dan memasang paket beserta dependensinya

Pembaruan

Memperbarui paket ke versi terbaru dari repository

Konfigurasi

Menyiapkan berkas konfigurasi awal secara otomatis

Penghapusan

Menghapus paket beserta komponen yang tak lagi diperlukan

APT (Debian/Ubuntu) & DNF (RHEL/CentOS)

APT — DEBIAN / UBUNTU

```
apt update
# memperbarui daftar paket
# dari repository

apt upgrade
# memperbarui seluruh paket
# yang terpasang

apt install nginx
# menginstal paket nginx

apt remove nginx
# menghapus paket nginx
```

YUM / DNF — RHEL / CENTOS / FEDORA

```
dnf check-update
# memeriksa pembaruan tersedia

dnf update
# memperbarui paket terpasang

dnf install httpd
# menginstal paket httpd

dnf remove httpd
# menghapus paket httpd
```

Windows Update dan PowerShell

Windows Server tidak menggunakan package manager berbasis repository seperti Linux, melainkan mekanisme pembaruan terpusat dari Microsoft.



Windows Update

Digunakan untuk pembaruan sistem operasi Windows Server, termasuk patch keamanan yang dirilis Microsoft secara berkala.



PowerShell — Install-WindowsFeature

Modul PowerShell untuk menambahkan peran (role) dan fitur server, misalnya peran web server (IIS) atau DNS Server.

Rangkuman 6.3 Package manager mengotomatisasi siklus hidup perangkat lunak — APT umum pada keluarga Debian, YUM/DNF pada keluarga Red Hat, dan Windows Update pada Windows Server.

Dependency dan Repository

Dependency

Sebuah paket sering membutuhkan paket lain agar dapat berjalan. Package manager modern menyelesaikan dependency ini secara otomatis — mengunduh dan memasang seluruh paket pendukung tanpa campur tangan manual.

Repository

Lokasi (server) yang menyimpan kumpulan paket beserta metadatanya. Dapat bersifat resmi (disediakan distribusi) maupun pihak ketiga.

MELIHAT DAFTAR REPOSITORY

```
cat /etc/apt/sources.list
# daftar repository pada
# Debian / Ubuntu
```

Administrator perlu berhati-hati menambahkan repository tidak resmi karena berisiko terhadap keamanan sistem.

Rangkuman 6.4 — Dependency memastikan kelengkapan komponen pendukung, sementara repository menjadi sumber terpercaya paket tersebut.

Jenis Pembaruan

Tidak semua pembaruan memiliki tingkat urgensi dan risiko yang sama — administrator perlu membedakannya saat menyusun jadwal patching.



Security Patch

Menambal celah keamanan yang ditemukan — prioritas tertinggi.



Bug Fix

Memperbaiki kesalahan fungsi perangkat lunak.



Feature Update

Menambahkan fitur baru pada perangkat lunak.



Major Upgrade

Perpindahan ke versi mayor baru — berisiko paling tinggi.

Praktik Patching Aman untuk Produksi

Melakukan patching langsung pada server produksi tanpa persiapan berisiko menyebabkan gangguan layanan. Praktik yang disarankan:

01

Uji di Staging

Menguji pembaruan pada lingkungan staging terlebih dahulu.

02

Backup Sebelum Update

Melakukan backup sebelum pembaruan (dibahas Bab 12).

03

Maintenance Window

Menjadwalkan pemeliharaan di luar jam sibuk.

04

Rolling Update

Menerapkan pembaruan bertahap pada banyak server, bukan sekaligus.

05

Rencana Rollback

Menyiapkan rencana pengembalian jika pembaruan bermasalah.

Rangkuman 6.5 — Strategi patching yang aman melibatkan pengujian, backup, penjadwalan, penerapan bertahap, dan rencana rollback.

Kegagalan Layanan Pasca-Pembaruan Sistem

Skenario Tim TI melakukan apt upgrade langsung pada server produksi tanpa pengujian. Setelah pembaruan, aplikasi utama perusahaan tidak dapat berjalan karena perubahan versi library yang tidak kompatibel.

Analisis Pembaruan dilakukan langsung tanpa lingkungan staging dan tanpa rencana rollback. Tim tidak memiliki cadangan konfigurasi maupun snapshot sistem, sehingga pemulihan memakan waktu lama.

Solusi Disarankan — Siklus Patching Terstruktur

Uji di Staging

Verifikasi kompatibilitas sebelum produksi

Snapshot / Backup

Cadangan sistem sebelum pembaruan produksi

Maintenance Window

Terjadwal dengan pemberitahuan kepada pengguna

Dokumentasi Versi

Agar rollback dapat dilakukan dengan cepat

Rangkuman

Package Manager

Package manager mengotomatisasi instalasi, pembaruan, dan penghapusan perangkat lunak pada server.

Dependency & Repository

Dependency memastikan kelengkapan komponen pendukung suatu paket, sementara repository menjadi sumber terpercaya paket tersebut.

Patching Terencana

Patching pada server produksi harus dilakukan secara terencana melalui pengujian, backup, penjadwalan, dan rencana rollback.

Pertanyaan Diskusi

Sebuah organisasi memiliki 20 server produksi yang harus diperbarui secara rutin setiap bulan.

Rancanglah strategi patching yang aman agar seluruh server dapat diperbarui tanpa menyebabkan gangguan layanan secara bersamaan.

Pertemuan Selanjutnya: Konfigurasi Layanan DNS Server

REFERENSI

Nemeth, E., et al. (2021). UNIX and Linux System Administration Handbook (5th ed.). Addison-Wesley.
Debian Project. APT User's Guide.
Red Hat. RHEL Documentation — DNF Package Manager.
Microsoft. Windows Server Update Services (WSUS) Documentation. Microsoft Learn.