

Final Report Keamanan Siber

Data Warehouse

Fadil Prasetyo Alfarizzi (123450048)

Reynaldi Rahmad (123450088)

Fathya Intami Gusda (123450095)

Melinza Nabila (123450122)



**PROGRAM STUDI SAINS DATA
FAKULTAS SAINS
2025**

MISI 1

DESAIN KONSEPTUAL DAN LOGIKAL

1.1. Business Requirements Analysis

Tujuan: Memahami kebutuhan bisnis dan analitik dari unit kerja keamanan siber untuk pengelolaan insiden keamanan (cyber incident) pada sistem jaringan organisasi.

1.1.1. Identifikasi Stakeholders

A. Pengguna utama data mart:

- Manajer Keamanan Informasi (CISO): memantau performa keamanan sistem dan tren serangan.
- Koordinator Tim Respon Insiden: mengevaluasi waktu respon dan efektivitas tim.
- Staff Keamanan IT: melakukan analisis teknis terhadap pola serangan dan titik rawan.
- Pimpinan Fakultas / Unit TI: memantau kondisi server dan prioritas mitigasi di masing-masing unit.

B. Decision Makers:

- Kepala Pusat Data & Keamanan TI.
- Rektor atau Wakil Rektor bidang TI (sebagai pengambil keputusan strategis).

1.1.2. Analisis Proses Bisnis

A. Proses bisnis utama yang terjadi dalam penanganan insiden keamanan meliputi:

- Deteksi Insiden oleh sistem monitoring.
- Klasifikasi Jenis Serangan (DDoS, Malware, SQL Injection, dll).
- Pencatatan Lokasi & Server terdampak.
- Respon oleh Tim Penanganan (Incident Response Team).
- Pelaporan hasil dan waktu penyelesaian.

B. Key Performance Indicators (KPI):

- Rata-rata durasi respon insiden.
- Jumlah insiden per bulan.
- Frekuensi serangan per fakultas/unit.
- Jumlah data yang terdampak.
- Persentase insiden yang diselesaikan tepat waktu.

C. Metrik Bisnis:

- Durasi respon (menit).
- Jumlah data terdampak (file/data).
- Status penyelesaian (selesai/pending).
- Frekuensi serangan (count per periode).

1.1.3. Kebutuhan Analitik

- A. Pertanyaan Bisnis yang perlu dijawab:
- Berapa banyak insiden keamanan yang terjadi setiap bulan?
 - Fakultas atau unit mana yang paling sering terkena serangan?
 - Jenis serangan apa yang paling sering terjadi?
 - Berapa lama rata-rata waktu yang dibutuhkan untuk menangani setiap insiden?
 - Siapa anggota tim yang paling cepat merespon serangan?
- B. Jenis Laporan yang Dibutuhkan:
- Laporan harian: daftar insiden yang baru terjadi dan status penyelesaiannya.
 - Laporan mingguan: tren jenis serangan dan aktivitas tim respon.
 - Laporan bulanan: evaluasi performa tim, jumlah serangan per lokasi, dan durasi respon rata-rata.
- C. Level Agregasi:
- Harian : Mingguan → Bulanan
 - Fakultas : Institusi
 - Jenis Serangan : Kategori Serangan

1.2. Conceptual Design

Tujuan:

Membuat model konseptual yang merepresentasikan entitas bisnis dan hubungan antar entitas pada sistem penanganan insiden keamanan.

1.2.1 Identifikasi Entitas

- A. Entitas utama:
- Incident : kejadian serangan siber yang terdeteksi.
 - Waktu : dimensi waktu terjadinya insiden.
 - Lokasi : unit/fakultas/server tempat insiden terjadi.
 - Jenis_Serangan : tipe serangan yang dialami sistem.

- Tim_Respon : tim yang menangani dan menutup insiden.

1.2.2 Atribut utama setiap entitas:

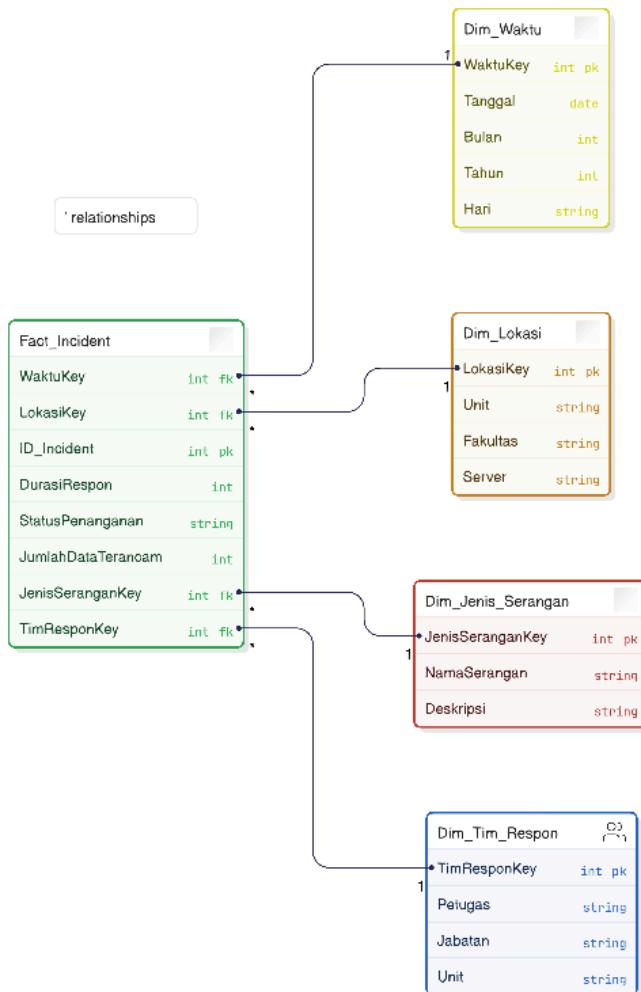
Entitas	Atribut	Primary/Foreign Key
Incident	ID_Incident, WaktuKey, LokasiKey, JenisSeranganKey, TimResponKey, DurasiRespon, StatusPenanganan, JumlahDataTerenam	PK: ID_Incident
Waktu	WaktuKey, Tanggal, Bulan, Tahun, Hari	PK: WaktuKey
Lokasi	LokasiKey, Fakultas, Server	PK: LokasiKey
Jenis_Serangan	JenisSeranganKey, NamaSerangan, Deskripsi	PK: JenisSeranganKey
Tim_Respon	TimResponKey, Petugas, Jabatan, Unit	PK: TimResponKey

1.2.3. Definisi Relationships

Relasi	Kardinalitas	Jenis
Waktu - Incident	1 – N	One-to-Many
Lokasi - Incident	1 – N	One-to-Many
Jenis_Serangan - Incident	1 – N	One-to-Many
Tim_Respon - Incident	1 – N	One-to-Many

Semua hubungan bersifat mandatory, karena setiap insiden harus memiliki waktu, lokasi, jenis serangan, dan tim respon yang menangani.

1.2.4. Gambar ERD



1.3. Dimensional Model

Tujuan:

Mengonversi model konseptual menjadi model dimensional untuk keperluan analisis pada data warehouse.

1.3. Identifikasi Fact Table

Fact Table: Fact_Incident

- Business Process: Pelaporan dan penanganan insiden keamanan.
- Grain: Satu record = satu kejadian insiden.
- Measures: DurasiRespon, JumlahDataTerenam.
- Additivity: Additive.

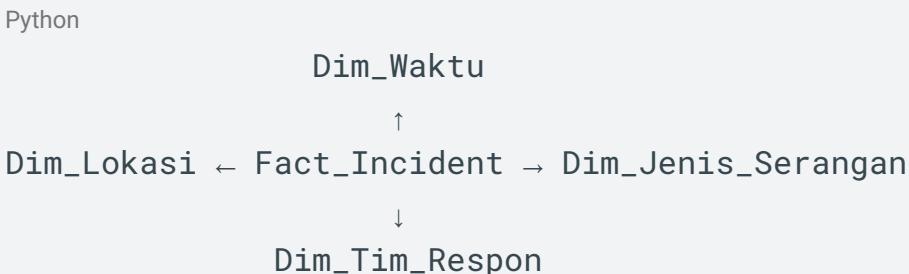
1.3.2. Identifikasi Dimension Tables

Dimensi	Fokus Analisis	Contoh Atribut
Dim_Waktu	Kapan insiden terjadi	Tanggal, Bulan, Tahun, Hari
Dim_Lokasi	Di mana insiden terjadi	Fakultas, Server
Dim_Jenis_Serangan	Apa jenis serangan	NamaSerangan, Deskripsi
Dim_Tim_Respon	Siapa yang menangani	Petugas, Jabatan, Unit

1.3.3. Desain Star Schema

Fact table berada di tengah, dikelilingi oleh empat dimensi.

Diagram:



Foreign Key Relationships:

- Fact_Incident.WaktuKey → Dim_Waktu.WaktuKey

- Fact_Incident.LokasiKey → Dim_Lokasi.LokasiKey
- Fact_Incident.JenisSeranganKey → Dim_Jenis_Serangan.JenisSeranganKey
- Fact_Incident.TimResponKey → Dim_Tim_Respon.TimResponKey

1.3.4. Penentuan Surrogate Keys

- Semua dimensi menggunakan integer surrogate key.
- Naming Convention:
Dim_Waktu_SK, Dim_Lokasi_SK, Dim_Jenis_Serangan_SK, Dim_Tim_Respon_SK
- Strategi Slowly Changing Dimensions (SCD):
Tipe 2 (History Tracking): bila data tim respon atau lokasi berubah, versi baru dibuat dengan SK baru agar histori tetap terjaga.

1.3.5. Data Dictionary

Tabel	Kolom	Tipe Data	Deskripsi
Fact_Incident	ID_Incident	int (PK)	Kunci utama insiden
	WaktuKey	int (FK)	Referensi ke Dim_Waktu
	LokasiKey	int (FK)	Referensi ke Dim_Lokasi
	JenisSeranganKey	int (FK)	Referensi ke Dim_Jenis_Serangan
	TimResponKey	int (FK)	Referensi ke Dim_Tim_Respon
	DurasiRespon	int	Lama penanganan insiden
	JumlahDataTerenam	int	Banyaknya data terdampak
	StatusPenanganan	string	Status: selesai/pending
Dim_Waktu	WaktuKey	int (PK)	Surrogate key waktu
	Tanggal	date	Tanggal kejadian

	Bulan	int	Bulan ke-n
	Tahun	int	Tahun kejadian
	Hari	string	Nama hari
Dim_Lokasi	LokasiKey	int (PK)	Surrogate key lokasi
	Fakultas	string	Fakultas terkait
	Server	string	Nama server terdampak
Dim_Jenis_Serangan	JenisSeranganKey	int (PK)	Surrogate key jenis serangan
	NamaSerangan	string	Nama jenis serangan
	Deskripsi	string	Penjelasan singkat
Dim_Tim_Respon	TimResponKey	int (PK)	Surrogate key tim respon
	Petugas	string	Nama petugas
	Jabatan	string	Jabatan petugas
	Unit	string	Unit kerja tim

MISI 2

DESAIN FISIKAL DAN DEVELOPMENT

2.1 Physical Database Design

2.1.1. Database Setup

Database diberi nama DM_KeamananSiber_DW dan terdiri atas:

- File data utama (Primary Data File)
- File data untuk tabel Data Warehouse (Secondary Data File)
- File log transaksi (Log File)

Struktur penyimpanan dibuat dengan ukuran awal tertentu dan aturan pertumbuhan (file growth) sehingga pengelolaan ruang disk lebih efisien.

Komponen File	Ukuran Awal	Maksimum	File Growth
Primary Data	512 MB	Unlimited	128 MB
Secondary Data	1024 MB	Unlimited	256 MB
Log File	256 MB	2 GB	64 MB

Pengaturan ini memberikan keseimbangan antara performansi dan efisiensi penyimpanan.

Agar database optimal sebagai Data Warehouse, diterapkan beberapa pengaturan:

- Recovery Model = SIMPLE
Untuk menghemat log serta memudahkan ETL.
- AUTO_SHRINK = OFF
Menghindari fragmentasi data.
- AUTO_CLOSE = OFF
Meningkatkan performa.
- READ_COMMITTED_SNAPSHOT = ON
Agar query untuk reporting tidak terjadi blocking.

Struktur Data :

Schema	Fungsi
staging	Penyimpanan data sementara sebelum diproses
dim	Menyimpan tabel dimensi
fact	Menyimpan tabel faktual
rw	Layer khusus untuk reporting (view)

2.1.2. Membuat Tabel Dimensi

A. Tabel Dim_Waktu

Atribut	Tipe Data	Deskripsi
WaktuKey	INT (PK)	Primary key yang merepresentasikan tanggal
Tanggal	DATE	Tanggal aktual
Bulan	TINYINT	Nilai bulan (1–12)
Tahun	SMALLINT	Nilai tahun
Hari	VARCHAR(15)	Nama hari (Senin–Minggu)
CreatedDate	DATETIME	Waktu pertama kali data dibuat
ModifiedDate	DATETIME	Waktu terakhir dimodifikasi

B. Tabel Dim_Lokasi

Atribut	Tipe Data	Deskripsi
LokasiKey	INT (PK, Identity)	Primary key
Fakultas	VARCHAR(50)	Nama fakultas/organisasi
Server	VARCHAR(100)	Nama server atau device
EffectiveDate	DATE	Tanggal data mulai berlaku
ExpiryDate	DATE	Tanggal data berakhir (null = aktif)
IsCurrent	BIT	Menandakan data aktif atau historis
CreatedDate	DATETIME	Timestamp pembuatan
ModifiedDate	DATETIME	Timestamp perubahan

C. Tabel Dim_Jenis_Serangan

Atribut	Tipe Data	Deskripsi
JenisSeranganKey	INT (PK, Identity)	Primary key
NamaSerangan	VARCHAR(100)	Nama jenis serangan
Deskripsi	VARCHAR(255)	Informasi tambahan
CreatedDate	DATETIME	Timestamp pembuatan
ModifiedDate	DATETIME	Timestamp perubahan

D. Tabel Dim_Tim_Respon

Atribut	Tipe Data	Deskripsi
TimResponKey	INT (PK, Identity)	Primary key
Petugas	VARCHAR(100)	Nama petugas yang menangani
Jabatan	VARCHAR(50)	Jabatan petugas
Unit	VARCHAR(50)	Unit organisasi
EffectiveDate	DATE	Tanggal mulai berlaku
ExpiryDate	DATE	Tanggal berakhir (null = aktif)
IsCurrent	BIT	Indikator aktif/historis
CreatedDate	DATETIME	Timestamp pembuatan
ModifiedDate	DATETIME	Timestamp perubahan

2.1.3. Membuat Tabel Fakta

A. Atribut Kunci

IncidentKey

- Tipe: BIGINT (Identity)
- Bersifat sebagai *primary key*
- Digunakan sebagai identitas unik untuk setiap kejadian insiden

B. Foreign Key

Tabel fakta terhubung ke tabel dimensi sebagai berikut:

- WaktuKey : Dim_Waktu
Menunjukkan kapan insiden terjadi.
- LokasiKey : Dim_Lokasi
Menunjukkan lokasi/ fakultas/ server yang diserang.
- JenisSeranganKey : Dim_Jenis_Serangan
Menunjukkan jenis serangan yang dialami.
- TimResponKey : Dim_Tim_Respon
Menunjukkan tim atau petugas yang menangani insiden.

C. Atribut Ukuran (Measures)

Parameter kuantitatif untuk analisis kinerja penanganan insiden:

- DurasiRespon : Lama penanganan insiden (menit/jam).
- JumlahDataTerenam : Jumlah data yang terdampak selama serangan.
- StatusPenanganan : Status seperti: *Open, In Progress, Closed*.

2.2. Indexing Strategy

2.2.1. Clustered Index

Clustered Index diterapkan untuk mempercepat proses akses data berdasarkan urutan waktu dan kejadian.

CIX_Fact_Incident_WaktuKey :

- Dibentuk pada atribut WaktuKey dan IncidentKey.
- Indeks ini mengatur data secara fisik dalam disk berdasarkan urutan waktu.

2.2.2. Non Clustered Index

Non-Clustered Index diterapkan untuk mempercepat proses pencarian berdasarkan atribut dimensi tertentu serta mendukung analisis statistik. Indeks ini memanfaatkan fitur **INCLUDE** untuk mengikutkan kolom yang sering diakses agar query lebih optimal.

Indeks yang dibuat adalah:

- A. IX_Fact_Incident_Lokasi
 - Dibentuk pada LokasiKey
 - Mengikutsertakan atribut DurasiRespon dan JumlahDataTerenam
 - Tujuan: Mempercepat analisis per lokasi (fakultas/server).
- B. IX_Fact_Incident_JenisSerangan
 - Dibentuk pada JenisSeranganKey
 - Mengikutsertakan DurasiRespon dan JumlahDataTerenam
 - Tujuan: Mempercepat analisis per jenis serangan.
- C. IX_Fact_Incident_TimRespon
 - Dibentuk pada TimResponKey
 - Mengikutsertakan DurasiRespon dan StatusPenanganan
 - Tujuan: Mendukung analisis kinerja tim respon.
- D. IX_Fact_Incident_Cover
 - Indeks gabungan (covering index) pada WaktuKey dan LokasiKey
 - Mengikutsertakan: JenisSeranganKey, DurasiRespon, JumlahDataTerenam, StatusPenanganan
 - Tujuan: Mempercepat query multidimensi yang menggabungkan tabel dimensi utama

2.2.3. Columnstore Index

Atribut yang digunakan:

- WaktuKey, LokasiKey, JenisSeranganKey, TimResponKey
- DurasiRespon, JumlahDataTerenam, StatusPenanganan

2.3. Partitioning Strategy

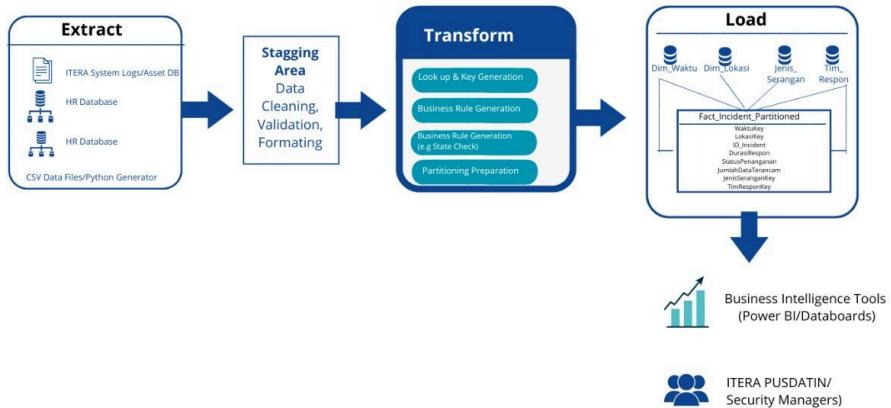
Partisi dilakukan menggunakan tipe data INT berdasarkan format YYYYMMDD

Rincian pembagian partisi :

Partisi	Periode Data
Partisi 1	Data < 1 Januari 2023
Partisi 2	2023
Partisi 3	2024
Partisi 4	2025 dan seterusnya

2.4 ETL Design

2.4.1. ETL Architecture Design



2.4.2. Staging Table

A. Staging Tabel Dimensi

- stg.Dim_Waktu : Berisi data atribut tanggal (hari, bulan, tahun) sebelum dimuat ke Dim_Waktu.
- stg.Dim_Lokasi : Berisi data lokasi seperti fakultas dan server tempat insiden terjadi.
- stg.Dim_JenisSerangan : Menampung kategori serangan (malware, phishing, brute force, dan sebagainya).
- stg.Dim_TimRespon : Berisi nama petugas, jabatan, dan unit yang menangani kejadian insiden.

B. Staging Tabel Fakta

- stg.Fact_Incident : Menampung data insiden sebelum diproses menjadi baris final dalam tabel Fact_Incident.
- Atribut yang disimpan mencakup:
 - 1) Identitas insiden
 - 2) Foreign key (dimensi)
 - 3) Durasi respon
 - 4) Jumlah data terdampak
 - 5) Status penanganan

C. ETL Mapping Document

Source	Source Column	Target	Target Column	Transformation
dbo.Dim_Waktu	WaktuKey	Dim_Waktu	WaktuKey	Direct mapping
dbo.Dim_Waktu	Tanggal	Dim_Waktu	Tanggal	Direct mapping
dbo.Dim_Waktu	Bulan	Dim_Waktu	Bulan	Direct mapping
dbo.Dim_Waktu	Tahun	Dim_Waktu	Tahun	Direct mapping
dbo.Dim_Waktu	Hari	Dim_Waktu	Hari	Direct mapping
dbo.Dim_Lokasi	LokasiKey	Dim_Lokasi	LokasiKey	Direct mapping
dbo.Dim_Lokasi	Fakultas	Dim_Lokasi	Fakultas	Direct mapping
dbo.Dim_Lokasi	Server	Dim_Lokasi	Server	Direct mapping
dbo.Dim_JenisSerangan	JenisSeranganKey	Dim_JenisSerangan	JenisSeranganKey	Direct mapping
dbo.Dim_JenisSerangan	NamaSerangan	Dim_JenisSerangan	NamaSerangan	Direct mapping
dbo.Dim_JenisSerangan	Deskripsi	Dim_JenisSerangan	Deskripsi	Direct mapping
dbo.Dim_TimRespon	TimResponKey	Dim_TimRespon	TimResponKey	Direct mapping
dbo.Dim_TimRespon	Petugas	Dim_TimRespon	Petugas	Direct mapping
dbo.Dim_TimRespon	Jabatan	Dim_TimRespon	Jabatan	Direct mapping
dbo.Dim_TimRespon	Unit	Dim_TimRespon	Unit	Direct mapping
dbo.Fact_Incident	ID_Incident	Fact_Incident	ID_Incident	Direct mapping
dbo.Fact_Incident	WaktuKey	Fact_Incident	WaktuKey	Direct mapping (FK sudah tersedia)
dbo.Fact_Incide	LokasiKey	Fact_Incident	LokasiKey	Direct mapping (FK

nt				sudah tersedia)
dbo.Fact_Incident	JenisSeranganKey	Fact_Incident	JenisSeranganKey	Direct mapping (FK sudah tersedia)
dbo.Fact_Incident	TimResponKey	Fact_Incident	TimResponKey	Direct mapping (FK sudah tersedia)
dbo.Fact_Incident	DurasiResponse	Fact_Incident	DurasiResponse	Direct mapping
dbo.Fact_Incident	JumlahDataTerenam	Fact_Incident	JumlahDataTerenam	Direct mapping
dbo.Fact_Incident	StatusPenanganan	Fact_Incident	StatusPenanganan	Direct mapping

2.5 ETL Implementation

A. Prosedur Load Dimensi

- Memuat data dimensi waktu dari *staging* ke tabel Dim_Waktu.
- Hanya menambahkan record baru yang belum pernah dimuat sebelumnya.
- Menghindari duplikasi data menggunakan WHERE NOT EXISTS.

B. Prosedur Load Fact

- Memuat data transaksi insiden keamanan ke tabel fact.
- Mengambil seluruh foreign key langsung dari staging (tidak perlu lookup lagi).
- Menstandarkan kolom StatusPenanganan ke huruf kapital (UPPER).
- Menghindari duplikasi data berdasarkan ID_Incident.

C. Prosedur Utama ETL

Tahap	Penjelasan
Load Dimensi	Memuat Dim_Waktu, Dim_Lokasi, Dim_JenisSerangan, Dim_TimRespon
Load Fact	Memuat Fact_Incident dari staging
Optimasi DB	sp_updatestats memperbarui statistik query optimizer
Transaksi	Menjamin data konsisten
Error Handling	Menggunakan TRY-CATCH & rollback jika error

2.6 Data Quality Assurance

A. Data Quality Checks

- Mendeteksi apakah ada kolom key, atribut penting, atau kolom fakta yang berisi nilai kosong (NULL).
- Mengecek apakah ada data pada tabel Fact_Incident yang merujuk ke Dimensi tetapi dimensinya tidak ada (broken / orphan reference).
- Memastikan DurasiRespon tidak memiliki nilai negatif.
- Memastikan StatusPenanganan hanya berisi nilai yang valid: *SELESAI* atau *PENDING*.
- Memastikan bahwa ID_Incident unik dan tidak terjadi duplikasi dalam tabel fakta.
- Mengecek apakah jumlah data pada staging area sudah sesuai dengan jumlah data pada data warehouse setelah proses ETL.

2.7 Performance Testing

- Menganalisis jumlah insiden berdasarkan jenis serangan dan status penanganannya pada tahun 2024.
- Melihat pola kejadian insiden berdasarkan Tahun, Bulan, Hari untuk mendeteksi tren jangka panjang maupun harian.
- Mengidentifikasi hubungan antara lokasi insiden, server, dan unit tim respon keamanan dalam menangani insiden.

MISI 3

IMPLEMENTASI PRODUKSI

3.1 Dashboard Development

A. Executive Summary Dashboard

- Banyak Kejadian Selama 3 Tahun

Memberikan ringkasan kondisi keamanan siber secara cepat dalam 3 tahun (2023–2025).

Dashboard ini dipakai untuk melihat:

- 1) Tren jumlah insiden tiap tahun
- 2) Rata-rata waktu respon insiden
- 3) Total insiden yang terjadi
- 4) Total data yang terdampak

- Jumlah Serangan Berdasarkan Jenis Serangan

Menunjukkan jenis serangan yang paling sering terjadi. Untuk :

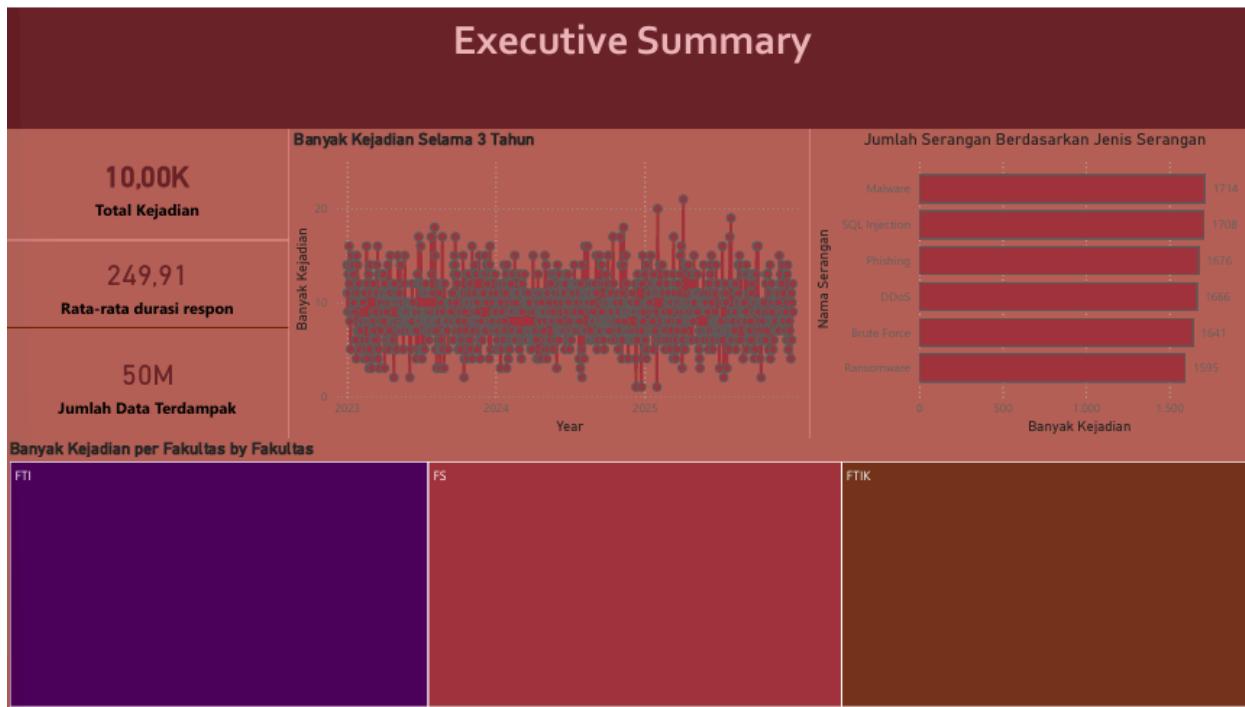
- 1) Menentukan prioritas proteksi (misalnya firewall, IDS, antivirus)
- 2) Melihat pola ancaman dominan

Jika Malware dan SQL Injection paling tinggi → fokus perbaikan di endpoint & database security.

- Banyak Kejadian per Fakultas

Menampilkan unit/fakultas mana yang paling sering terkena serangan. Dashboard ini dipakai manajemen untuk:

- 1) Mengetahui unit paling rentan
- 2) Menentukan prioritas audit keamanan



B. Operational Performance Dashboard

- Petugas - Rata-rata Waktu Respon

Menilai performa tiap petugas dalam menangani insiden. Digunakan untuk:

- 1) Evaluasi kinerja tim
- 2) Penentuan pelatihan atau pembagian tugas

- Heatmap Rata-rata Respon per Server & Hari

Menunjukkan pola performa berdasarkan hari dan server. Membantu menjawab:

- 1) Hari apa respon paling lambat?
- 2) Server mana yang paling bermasalah?

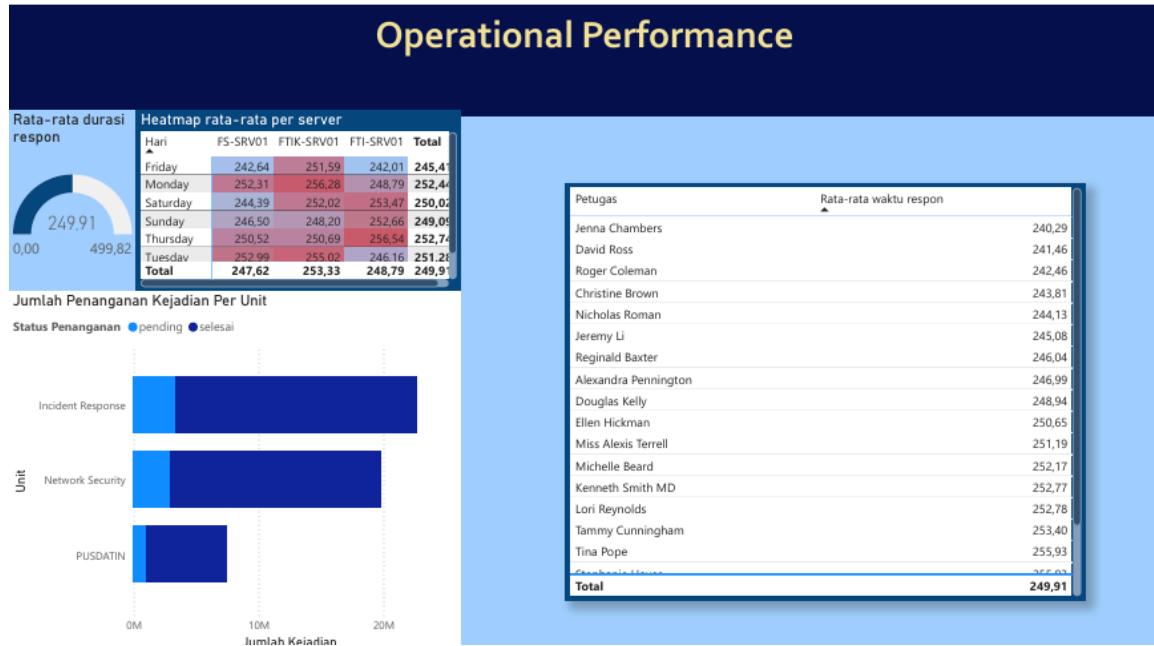
- Jumlah Penanganan per Unit

Menampilkan beban kerja tiap unit (Incident Response, Network Security, PUSDATIN). Digunakan untuk resource planning:

- 1) Perlu tambah personel?

- 2) Unit mana overload?
- Status Penanganan (Pending vs Selesai)

Memantau tingkat penyelesaian insiden. Jika banyak “pending” maka menunjukkan bottleneck proses.



C. Impact Analysis Dashboard

- Total Dampak Data per Tahun

Melihat skala kerusakan/dampak serangan. Digunakan untuk:

- 1) Estimasi risiko
- 2) Justifikasi anggaran keamanan

- Distribusi Dampak per Jenis Serangan

Melihat jenis serangan yang paling merugikan. Kadang serangan jarang terjadi tapi dampaknya besar.

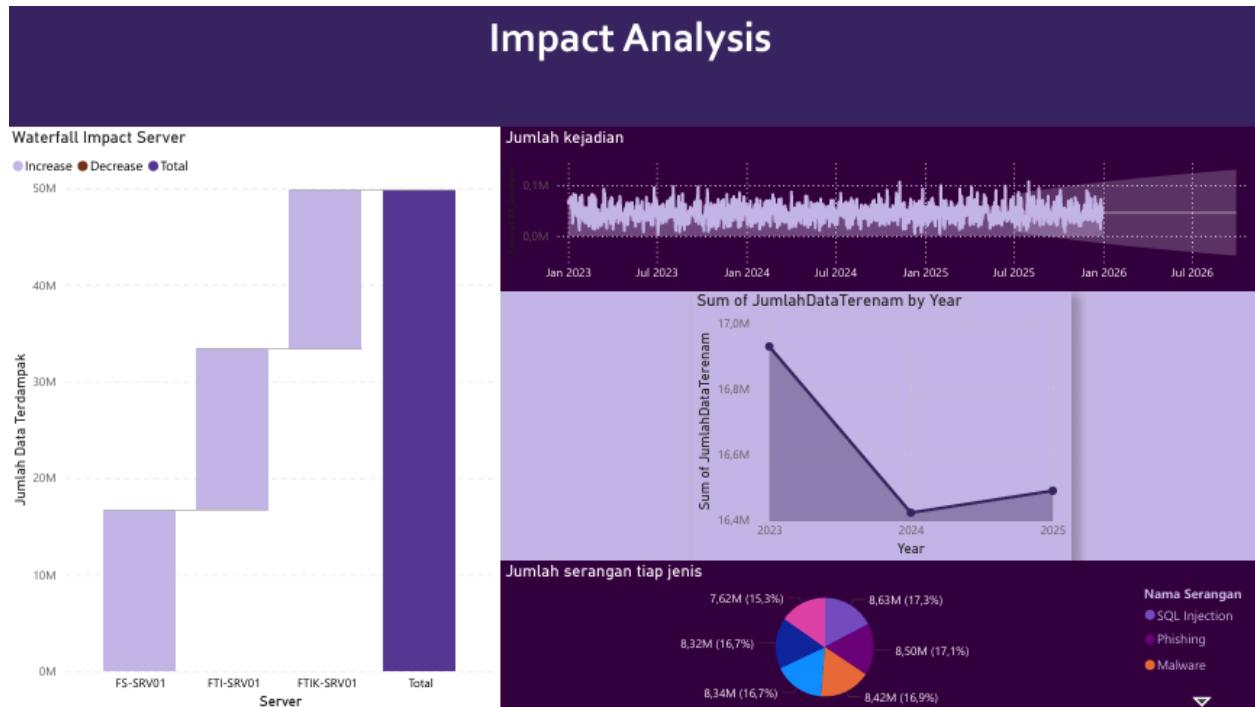
- Waterfall Impact Server

Mengukur kontribusi tiap server terhadap total kerusakan data. Membantu fokus patching & hardening server bermasalah.

- Timeline Jumlah Insiden

Melihat tren waktu kejadian insiden (time series). Untuk:

- 1) Mendeteksi pola musiman
- 2) Deteksi lonjakan serangan



3.2 Security Implementation

3.2.1. User Roles

Roles yang digunakan:

- db_etl_operator
- db_executive
- db_analyst
- db_viewer

3.2.2. Users and Assign Roles

A. ETL Operator

Hak akses yang diberikan:

- EXECUTE: Menjalankan stored procedure ETL utama
usp_Master_ETL_KeamananSiber
- SELECT, INSERT, UPDATE, DELETE pada skema stg untuk extract, cleaning, dan prepare data
- INSERT pada skema dbo : Untuk memindahkan data bersih dari staging ke tabel fakta & dimensi

B. Executive

Hak akses yang diberikan:

- SELECT pada seluruh tabel skema dbo untuk membaca data manajemen & laporan strategis
- EXECUTE pada stored procedure ETL untuk monitoring atau menjalankan ulang ETL dalam keadaan darurat

C. Analyst

Hak akses yang diberikan :

- SELECT di skema dbo (data final untuk analisis)
- FULL ACCESS (SELECT, INSERT, UPDATE, DELETE) di skema stg untuk transformasi data, cleaning tambahan, dan preprocessing

D. Viewer

Hak Akses yang diberikan :

- SELECT pada:
 - 1) Fact_Incident
 - 2) Dim_Waktu
 - 3) Dim_Lokasi

3.3 Backup and Recovery Strategy

Melakukan pencadangan penuh (*full backup*) database dengan tujuan menjaga keamanan data, memastikan ketersediaan cadangan apabila terjadi kerusakan, dan mempermudah proses pemulihan sistem.

Parameter :

1. COMPRESSION

Untuk memperkecil ukuran file backup, kompresi menjaga efisiensi ruang penyimpanan tanpa mengurangi integritas data.

2. INIT

Membuat SQL Server menimpa file backup yang sudah ada sebelumnya. Maka file hasil backup selalu berisi cadangan terbaru sehingga menghindari penumpukan file backup lama.

3. NAME = 'Full Database Backup'

Memberikan nama atau label deskriptif pada hasil backup, agar mudah dikenali saat proses restore atau verifikasi backup dilakukan.

4. STATS = 10

Menampilkan progres proses pencadangan pada setiap 10%. Untuk mempermudah pemantauan status backup terutama untuk database berukuran besar.

3.4 User Acceptance Testing

Tujuan : Validasi sistem dengan end - user

Aktivitas

1. Create Test Cases

Test id	Scenario	Expected Result	Status	Notes
TC001	Menampilkan Status Kejadian yang Berstatus PENDING	menampilkan insiden dengan StatusPenanganan = "PENDING"	pass	-

2. Conduct UAT Session

- Schedule meeting dengan stakeholders
- Demonstrasi dashboard dan fitur
- Collect feedback dan bug reports
- Document change requests

3. Performance Testing

- Test dengan concurrent users
- Measure dashboard load time
- Test ETL dengan production data volume
- Identify and fix bottlenecks

4. Refinement

- Fix bugs dari UAT
- Implement requested features
- Optimize slow queries
- Update documentation

LAMPIRAN

SQL

a. Database Setup

```
SQL
-- Create Database
CREATE DATABASE DM_KeamananSiber_DW
ON PRIMARY
(
    NAME = N'DM_KeamananSiber_DW_Data',
    FILENAME = N'D:\Data\DM_KeamananSiber_DW_Data.mdf',
    SIZE = 1GB,
    MAXSIZE = UNLIMITED,
    FILEGROWTH = 256MB
)
LOG ON
(
    NAME = N'DM_KeamananSiber_DW_Log',
    FILENAME = N'E:\Logs\DM_KeamananSiber_DW_Log.ldf',
    SIZE = 256MB,
    MAXSIZE = 2GB,
    FILEGROWTH = 64MB
);
GO

USE DM_KeamananSiber_DW;
GO
```

b. Tabel Dimensi

```
SQL
Tabel Dimensi
CREATE TABLE dbo.Dim_Waktu (
    WaktuKey INT PRIMARY KEY NOT NULL,
    Tanggal DATE NOT NULL,
    Bulan TINYINT NOT NULL,
    Tahun SMALLINT NOT NULL,
    Hari VARCHAR(15) NOT NULL,
```

```
        CreatedDate DATETIME DEFAULT GETDATE(),
        ModifiedDate DATETIME DEFAULT GETDATE()
    );
GO

CREATE TABLE dbo.Dim_Lokasi (
    LokasiKey INT IDENTITY(1,1) PRIMARY KEY NOT NULL,
    Fakultas VARCHAR(50) NOT NULL,
    Server VARCHAR(100) NOT NULL,

    -- Metadata
    EffectiveDate DATE NOT NULL DEFAULT GETDATE(),
    ExpiryDate DATE NULL,
    IsCurrent BIT NOT NULL DEFAULT 1,
    CreatedDate DATETIME DEFAULT GETDATE(),
    ModifiedDate DATETIME DEFAULT GETDATE()
);
GO

CREATE TABLE dbo.Dim_Jenis_Serangan (
    JenisSeranganKey INT IDENTITY(1,1) PRIMARY KEY NOT NULL,
    NamaSerangan VARCHAR(100) NOT NULL,
    Deskripsi VARCHAR(255) NULL,

    CreatedDate DATETIME DEFAULT GETDATE(),
    ModifiedDate DATETIME DEFAULT GETDATE()
);
GO

CREATE TABLE dbo.Dim_Jenis_Serangan (
    JenisSeranganKey INT IDENTITY(1,1) PRIMARY KEY NOT NULL,
    NamaSerangan VARCHAR(100) NOT NULL,
    Deskripsi VARCHAR(255) NULL,

    CreatedDate DATETIME DEFAULT GETDATE(),
    ModifiedDate DATETIME DEFAULT GETDATE()
);
GO

CREATE TABLE dbo.Dim_Tim_Respon (
    TimResponKey INT IDENTITY(1,1) PRIMARY KEY NOT NULL,
    Petugas VARCHAR(100) NOT NULL,
    Jabatan VARCHAR(50) NOT NULL,
```

```

Unit VARCHAR(50) NOT NULL,
-- SCD2
EffectiveDate DATE NOT NULL DEFAULT GETDATE(),
ExpiryDate DATE NULL,
IsCurrent BIT NOT NULL DEFAULT 1,
-- Metadata
CreatedDate DATETIME DEFAULT GETDATE(),
ModifiedDate DATETIME DEFAULT GETDATE()
);
GO

```

c. Tabel Fakta

```

SQL
-- Fact Table: Fact_Incident
CREATE TABLE dbo.Fact_Incident (
    IncidentKey BIGINT IDENTITY(1,1) PRIMARY KEY NOT NULL,
    -- Foreign Keys
    WaktuKey INT NOT NULL,
    LokasiKey INT NOT NULL,
    JenisSeranganKey INT NOT NULL,
    TimResponKey INT NOT NULL,
    -- Degenerate Dimension
    IncidentCode VARCHAR(50) NULL,
    -- Measures
    DurasiRespon INT NOT NULL,
    JumlahDataTerenam INT NOT NULL,
    StatusPenanganan VARCHAR(20) NOT NULL,
    -- Metadata
    SourceSystem VARCHAR(50) NOT NULL DEFAULT 'SYNTHETIC_GENERATOR',
    LoadDate DATETIME DEFAULT GETDATE(),
    -- FK Constraints
    CONSTRAINT FK_FactIncident_Waktu

```

```

    FOREIGN KEY (WaktuKey) REFERENCES dbo.Dim_Waktu(WaktuKey),

CONSTRAINT FK_FactIncident_Lokasi
    FOREIGN KEY (LokasiKey) REFERENCES dbo.Dim_Lokasi(LokasiKey),

CONSTRAINT FK_FactIncident_JSerangan
    FOREIGN KEY (JenisSeranganKey) REFERENCES
dbo.Dim_Jenis_Serangan(JenisSeranganKey),

CONSTRAINT FK_FactIncident_TimRespon
    FOREIGN KEY (TimResponKey) REFERENCES dbo.Dim_Tim_Respon(TimResponKey)
);
GO

```

d. Index

```

SQL
CREATE CLUSTERED INDEX CIX_Fact_Enrollment_DateKey
ON dbo.Fact_Enrollment(DateKey, EnrollmentKey);

-- Clustered Index
CREATE CLUSTERED INDEX CIX_Fact_Incident_WaktuKey
ON dbo.Fact_Incident(WaktuKey, IncidentKey);
GO

CREATE NONCLUSTERED INDEX IX_Fact_Incident_Lokasi
ON dbo.Fact_Incident(LokasiKey)
INCLUDE (DurasiRespon, JumlahDataTerenam);
GO

CREATE NONCLUSTERED INDEX IX_Fact_Incident_JenisSerangan
ON dbo.Fact_Incident(JenisSeranganKey)
INCLUDE (DurasiRespon, JumlahDataTerenam);
GO

CREATE NONCLUSTERED INDEX IX_Fact_Incident_TimRespon
ON dbo.Fact_Incident(TimResponKey)
INCLUDE (DurasiRespon, StatusPenanganan);
GO

```

```

CREATE NONCLUSTERED INDEX IX_Fact_Incident_Cover
ON dbo.Fact_Incident(WaktuKey, LokasiKey)
INCLUDE (JenisSeranganKey, DurasiRespon, JumlahDataTerenam, StatusPenanganan);
GO

CREATE NONCLUSTERED COLUMNSTORE INDEX NCCIX_Fact_Incident
ON dbo.Fact_Incident
(
    WaktuKey,
    LokasiKey,
    JenisSeranganKey,
    TimResponKey,
    DurasiRespon,
    JumlahDataTerenam,
    StatusPenanganan
);
GO

```

e. Partition

SQL

```

-- Create Partition Function (by Incident Year)
CREATE PARTITION FUNCTION PF_IncidentYear (INT)
AS RANGE RIGHT FOR VALUES
(
    20230101, -- Tahun 2023
    20240101, -- Tahun 2024
    20250101 -- Tahun 2025
);
GO
-- Create Partition Scheme
CREATE PARTITION SCHEME PS_IncidentYear
AS PARTITION PF_IncidentYear
ALL TO ([PRIMARY]);
GO

```

f. Staging

SQL

```
-- Staging Schema
CREATE SCHEMA stg;
GO

-- Staging Table for Dim_Waktu
CREATE TABLE stg.Dim_Waktu (
    WaktuKey INT,
    Tanggal DATE,
    Bulan INT,
    Tahun INT,
    Hari VARCHAR(20),
    LoadDate DATETIME DEFAULT GETDATE()
);
GO

-- Staging Table for Dim_Lokasi
CREATE TABLE stg.Dim_Lokasi (
    LokasiKey INT,
    Fakultas VARCHAR(50),
    Server VARCHAR(50),
    LoadDate DATETIME DEFAULT GETDATE()
);
GO

-- Staging Table for Dim_JenisSerangan
CREATE TABLE stg.Dim_JenisSerangan (
    JenisSeranganKey INT,
    NamaSerangan VARCHAR(100),
    Deskripsi VARCHAR(255),
    LoadDate DATETIME DEFAULT GETDATE()
);
GO

-- Staging Table for Dim_TimRespon
CREATE TABLE stg.Dim_TimRespon (
    TimResponKey INT,
    Petugas VARCHAR(150),
    Jabatan VARCHAR(50),
    Unit VARCHAR(50),
    LoadDate DATETIME DEFAULT GETDATE()
);
GO
```

```
-- Staging Table for Fact_Incident
CREATE TABLE stg.Fact_Incident (
    ID_Incident INT,
    WaktuKey INT,
    LokasiKey INT,
    JenisSeranganKey INT,
    TimResponKey INT,
    DurasiRespon INT,
    JumlahDataTerenam INT,
    StatusPenanganan VARCHAR(20),
    LoadDate DATETIME DEFAULT GETDATE()
);
GO
```

g. ETL Procedure

```
SQL
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO

-- 1. PROSEDUR LOAD DIMENSI (DIMENSIONS LOAD PROCEDURES)
-- (Dihilangkan untuk fokus pada perbaikan Fact Load)

-- Prosedur: Load Dim_Waktu (Type 1 Load / Pre-Loaded)
CREATE OR ALTER PROCEDURE dbo.usp_Load_Dim_Waktu
AS
BEGIN
    SET NOCOUNT ON;

    INSERT INTO dbo.Dim_Waktu (WaktuKey, Tanggal, Bulan, Tahun, Hari)
    SELECT s.WaktuKey, s.Tanggal, s.Bulan, s.Tahun, s.Hari
    FROM stg.Dim_Waktu s
    WHERE NOT EXISTS (SELECT 1 FROM dbo.Dim_Waktu dw WHERE dw.WaktuKey =
s.WaktuKey);
END;
GO
```

```
-- Prosedur: Load Dim_Lokasi, Dim_JenisSerangan, Dim_TimRespon (Asumsi sudah dibuat)
-- ... (Prosedur SCD Type 2 lainnya diabaikan di sini)
GO

--
=====
== 
-- 2. PROCEDURE LOAD FACT TABLE (SOLUSI TANPA ERROR KOLOM)
-- Asumsi: stg.Fact_Incident sudah memiliki semua Foreign Key
-- 
=====

==

CREATE OR ALTER PROCEDURE dbo.usp_Load_Fact_Incident
AS
BEGIN
    SET NOCOUNT ON;

    INSERT INTO dbo.Fact_Incident (
        ID_Incident, WaktuKey, LokasiKey, JenisSeranganKey, TimResponKey,
        DurasiRespon, JumlahDataTerenam, StatusPenanganan,
        CreatedDate, ModifiedDate
    )
    SELECT
        s.ID_Incident,
        s.WaktuKey, -- AMBIL LANGSUNG DARI STAGING
        s.LokasiKey, -- AMBIL LANGSUNG DARI STAGING
        s.JenisSeranganKey, -- AMBIL LANGSUNG DARI STAGING
        s.TimResponKey, -- AMBIL LANGSUNG DARI STAGING

        s.DurasiRespon,
        s.JumlahDataTerenam,
        UPPER(s.StatusPenanganan),
        GETDATE(),
        GETDATE()
    FROM stg.Fact_Incident s

    -- Mencegah duplikasi Fact
    WHERE NOT EXISTS (
        SELECT 1
        FROM dbo.Fact_Incident f
        WHERE f.ID_Incident = s.ID_Incident
    );

```

```
END;
GO

-- 
=====

==

-- 3. MASTER ETL PROCEDURE
-- 
=====

==

CREATE OR ALTER PROCEDURE dbo.usp_Master_ETL_KeamananSiber
AS
BEGIN
    BEGIN TRY
        BEGIN TRANSACTION;

        -- Step 1: Load Dimensions
        EXEC dbo.usp_Load_Dim_Waktu;
        EXEC dbo.usp_Load_Dim_Lokasi;
        EXEC dbo.usp_Load_Dim_JenisSerangan;
        EXEC dbo.usp_Load_Dim_TimRespon;

        -- Step 2: Load Facts
        EXEC dbo.usp_Load_Fact_Incident;

        -- Step 3: Optimization & Finalization
        EXEC sp_updatestats;

        COMMIT TRANSACTION;

        PRINT 'ETL Keamanan Siber Completed Successfully';
    END TRY
    BEGIN CATCH
        IF @@TRANCOUNT > 0
            ROLLBACK TRANSACTION;

        DECLARE @ErrorMessage NVARCHAR(4000) = ERROR_MESSAGE();
        THROW;
    END CATCH
END;
GO
```

h. Quality Check

SQL

```
-- Pastikan Anda menjalankan script ini dengan koneksi ke database
DM_KeamananSiber_DW

-- =====
-- Check 1: Completeness - NULL values (Cek Dimensi & Fact)
-- =====

SELECT
    'Dim_Lokasi' AS TableName,
    COUNT(*) AS TotalRows,
    SUM(CASE WHEN LokasiKey IS NULL THEN 1 ELSE 0 END) AS NullLokasiKey,
    SUM(CASE WHEN Fakultas IS NULL THEN 1 ELSE 0 END) AS NullFakultas,
    SUM(CASE WHEN Server IS NULL THEN 1 ELSE 0 END) AS NullServer
FROM dbo.Dim_Lokasi
UNION ALL
SELECT
    'Fact_Incident' AS TableName,
    COUNT(*) AS TotalRows,
    SUM(CASE WHEN ID_Incident IS NULL THEN 1 ELSE 0 END) AS NullIDIncident,
    SUM(CASE WHEN DurasiRespon IS NULL THEN 1 ELSE 0 END) AS NullDurasi,
    SUM(CASE WHEN StatusPenanganan IS NULL THEN 1 ELSE 0 END) AS NullStatus
FROM dbo.Fact_Incident;

---

-- =====
-- Check 2: Consistency - Referential Integrity (Orphan Records)
-- =====

SELECT
    'TimResponKey' AS OrphanKey,
    COUNT(*) AS OrphanRecords
FROM dbo.Fact_Incident f
LEFT JOIN dbo.Dim_TimRespon dtr ON f.TimResponKey = dtr.TimResponKey
WHERE dtr.TimResponKey IS NULL -- Baris Fact yang tidak punya Tim Respon yang
valid
UNION ALL
SELECT
    'LokasiKey' AS OrphanKey,
    COUNT(*) AS OrphanRecords
FROM dbo.Fact_Incident f
LEFT JOIN dbo.Dim_Lokasi dl ON f.LokasiKey = dl.LokasiKey
WHERE dl.LokasiKey IS NULL;
```

```

---

-- =====
-- Check 3: Accuracy - Valid Ranges/Domain (Durasi dan Status)
-- =====
SELECT
    'Fact_Incident' AS TableName,
    SUM(CASE WHEN DurasiRespon < 0 THEN 1 ELSE 0 END) AS InvalidDurasiRespon,
    -- Durasi tidak boleh negatif
    SUM(CASE WHEN StatusPenanganan NOT IN ('SELESAI', 'PENDING') THEN 1 ELSE 0
END) AS InvalidStatusDomain -- Cek domain status
FROM dbo.Fact_Incident;

---

-- =====
-- Check 4: Duplicates (Memastikan ID_Incident unik)
-- =====
SELECT
    ID_Incident,
    COUNT(*) AS DuplicateCount
FROM dbo.Fact_Incident
GROUP BY ID_Incident
HAVING COUNT(*) > 1;

---

-- =====
-- Check 5: Record Counts Reconciliation (Membandingkan jumlah baris Staging vs
Warehouse)
-- =====
SELECT
    'Staging' AS DataSource,
    COUNT(*) AS RecordCount
FROM stg.Fact_Incident -- DIGANTI ke stg.Fact_Incident untuk menghindari Error
208
UNION ALL
SELECT
    'Warehouse' AS DataSource,
    COUNT(*) AS RecordCount
FROM dbo.Fact_Incident;

```

i. Test Query

```
SQL
-- Perintah ini diaktifkan untuk menganalisis kinerja query
SET STATISTICS TIME ON;
SET STATISTICS IO ON;

-- =====
-- Query 1: Total Insiden per Jenis Serangan dan Status (Modifikasi Query 1 Template)
-- =====

SELECT
    djs.NamaSerangan,
    f.StatusPenanganan,
    COUNT(f.ID_Incident) AS TotalIncidents,
    AVG(f.DurasiRespon) AS AvgResponseDuration,
    SUM(f.JumlahDataTerenam) AS TotalDataExposed
FROM dbo.Fact_Incident f
INNER JOIN dbo.Dim_JenisSerangan djs ON f.JenisSeranganKey =
djs.JenisSeranganKey
INNER JOIN dbo.Dim_Waktu dw ON f.WaktuKey = dw.WaktuKey
WHERE dw.Tahun = 2024 -- Batasi analisis pada tahun tertentu
GROUP BY djs.NamaSerangan, f.StatusPenanganan
ORDER BY TotalIncidents DESC;

---

-- =====
-- Query 2: Tren Insiden Bulanan/Harian (Modifikasi Query 2 Template)
-- =====

SELECT
    dw.Tahun,
    dw.Bulan,
    dw.Hari AS HariKejadian,
    COUNT(f.ID_Incident) AS TotalIncidents,
    AVG(f.DurasiRespon) AS AvgDuration
FROM dbo.Fact_Incident f
INNER JOIN dbo.Dim_Waktu dw ON f.WaktuKey = dw.WaktuKey
GROUP BY dw.Tahun, dw.Bulan, dw.Hari
ORDER BY dw.Tahun, dw.Bulan, dw.Hari;

---
```

```
-- =====
-- Query 3: Analisis Lokasi dan Tim Respon
-- =====

SELECT
    dl.Fakultas,
    dl.Server,
    dtr.Unit AS UnitRespon,
    COUNT(f.ID_Incident) AS TotalIncidents,
    AVG(f.DurasiRespon) AS AvgDuration
FROM dbo.Fact_Incident f
INNER JOIN dbo.Dim_Lokasi dl ON f.LokasiKey = dl.LokasiKey
INNER JOIN dbo.Dim_TimRespon dtr ON f.TimResponKey = dtr.TimResponKey
GROUP BY dl.Fakultas, dl.Server, dtr.Unit
ORDER BY TotalIncidents DESC;
```

j. Security

SQL

```
USE [DM_KeamananSiber_DW];
GO

-- CATATAN: Perintah CREATE ROLE dilewati karena Roles sudah ada di database.

-- =====
-- Grant Permissions for ETL Operator (Izin untuk Proses ETL)
-- =====
-- Memberikan hak EXECUTE untuk menjalankan Master Stored Procedure
GRANT EXECUTE ON dbo.usp_Master_ETL_KeamananSiber TO db_etl_operator;
-- Memberikan hak penuh pada Staging Area (untuk memuat dan membersihkan data)
GRANT SELECT, INSERT, UPDATE, DELETE ON SCHEMA::stg TO db_etl_operator;
-- Memberikan hak INSERT pada skema dbo (untuk memuat data ke Fact/Dimensi)
GRANT INSERT ON SCHEMA::dbo TO db_etl_operator;
GO

-- =====
-- Grant Permissions for Executive (Izin untuk Pengambilan Keputusan Tingkat Tinggi)
-- =====
-- Memberikan hak SELECT pada skema dbo (semua data final di DW)
```

```

GRANT SELECT ON SCHEMA::dbo TO db_executive;
-- Memberikan hak untuk menjalankan Master SP (untuk pengawasan atau eksekusi darurat)
GRANT EXECUTE ON dbo.usp_Master_ETL_KeamananSiber TO db_executive;
GO

-- =====
-- Grant Permissions for Analyst (Izin untuk Analisis dan Transformasi Data)
-- =====
-- Memberikan hak SELECT pada skema dbo (untuk melakukan analisis pada data final)
GRANT SELECT ON SCHEMA::dbo TO db_analyst;
-- Memberikan hak penuh pada Staging Area (untuk membersihkan dan memanipulasi data sebelum Load)
GRANT SELECT, INSERT, UPDATE, DELETE ON SCHEMA::stg TO db_analyst;
GO

-- =====
-- Grant Permissions for Viewer (Izin Baca-Saja pada View Analitik & Dimensi Dasar)
-- (Disini menggunakan tabel dasar yang pasti ada, karena Views sebelumnya tidak ditemukan)
-- =====
GRANT SELECT ON dbo.Fact_Incident TO db_viewer; -- Akses ke Fact Table
GRANT SELECT ON dbo.Dim_Waktu TO db_viewer;
GRANT SELECT ON dbo.Dim_Lokasi TO db_viewer;
GO

```

k. Recovery

SQL

```

-- Full Backup
BACKUP DATABASE DM_KeamananSiber_DW
TO DISK = N'C:\Backup\DM_KeamananSiber_DW_Full.bak'
WITH
    COMPRESSION, -- Mengurangi ukuran file backup
    INIT,        -- Menimpa file backup yang ada
    NAME = N'Full Database Backup',
    STATS = 10;
GO

```

ETL

a. Dim_Jenis_Serangan

JenisSeranganKey	NamaSerangan	Deskripsi
1	DDoS	Listen one next measure oil pattern produce either.
2	SQL Injection	Remain week develop music certainly.
3	Phishing	College turn society number.
4	Malware	Wife trip area increase hand development write others.
5	Brute Force	Material theory agency race off upon performance company
6	Ransomware	Those between indeed child.

b. Dim_Lokasi

LokasiKey	Fakultas	Server
1	FS	FS-SRV01
2	FTI	FTI-SRV01
3	FTIK	FTIK-SRV01

c. Dim_Tim_Respon

TimResponKey	Petugas	Jabatan	Unit
1	Tina Pope	Koordinator	PUSDATIN
2	Cory Juarez MD	Staff	PUSDATIN
3	Jamie Ortega	Supervisor	PUSDATIN
4	Kenneth Smith MD	Koordinator	Network Security
5	Roger Coleman	Koordinator	Network Security
6	David Ross	Koordinator	Incident Response
7	Lori Reynolds	Staff	Network Security
8	Miss Alexis Terrell	Staff	Network Security
9	Nicholas Roman	Staff	Incident Response
10	Alexandra Pennington	Koordinator	Incident Response
11	Stephanie Hayes	Supervisor	Incident Response
12	Ellen Hickman	Koordinator	Incident Response
13	Tammy Cunningham	Supervisor	Incident Response
14	Jenna Chambers	Koordinator	Incident Response
15	Douglas Kelly	Koordinator	Network Security
16	Don Murray	Koordinator	Network Security
17	Michelle Beard	Staff	Network Security
18	Jeremy Li	Supervisor	Network Security
19	Reginald Baxter	Staff	Incident Response
20	Christine Brown	Koordinator	Incident Response

d. Dim_Waktu

WaktuKey	Tanggal	Bulan	Tahun	Hari
1	2023-01-01	1	2023	Sunday
2	2023-01-02	1	2023	Monday
3	2023-01-03	1	2023	Tuesday
4	2023-01-04	1	2023	Wednesday
5	2023-01-05	1	2023	Thursday
6	2023-01-06	1	2023	Friday
7	2023-01-07	1	2023	Saturday
8	2023-01-08	1	2023	Sunday
9	2023-01-09	1	2023	Monday
10	2023-01-10	1	2023	Tuesday

Menampilkan 10 baris teratas

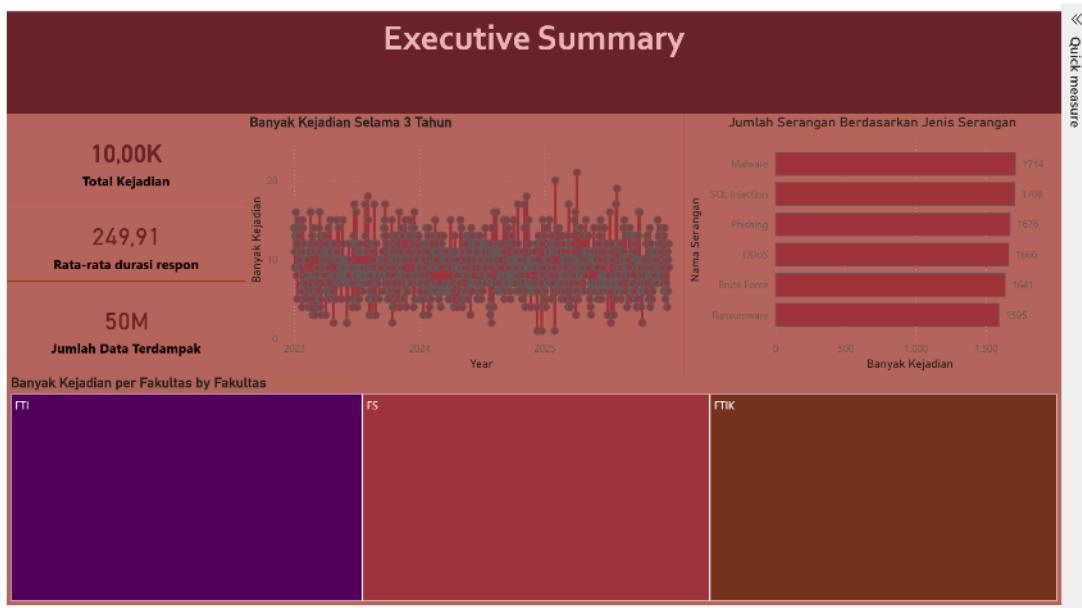
e. Fact_Incident

ID_Incident	WaktuKey	LokasiKey	JenisSeranganKey	TimResponKey	DurasiRespon	JumlahDataTerenam	StatusPenanganan
1	519	3	5	9	352	2696	selesai
2	136	1	1	14	357	2821	selesai
3	334	2	6	19	10	5615	pending
4	381	1	1	2	55	1490	selesai
5	999	3	1	20	13	6615	selesai
6	101	3	6	2	329	827	selesai
7	553	3	6	12	248	3823	pending
8	490	3	5	8	39	9795	selesai
9	648	1	2	7	140	4099	selesai
10	878	2	6	16	395	9206	pending

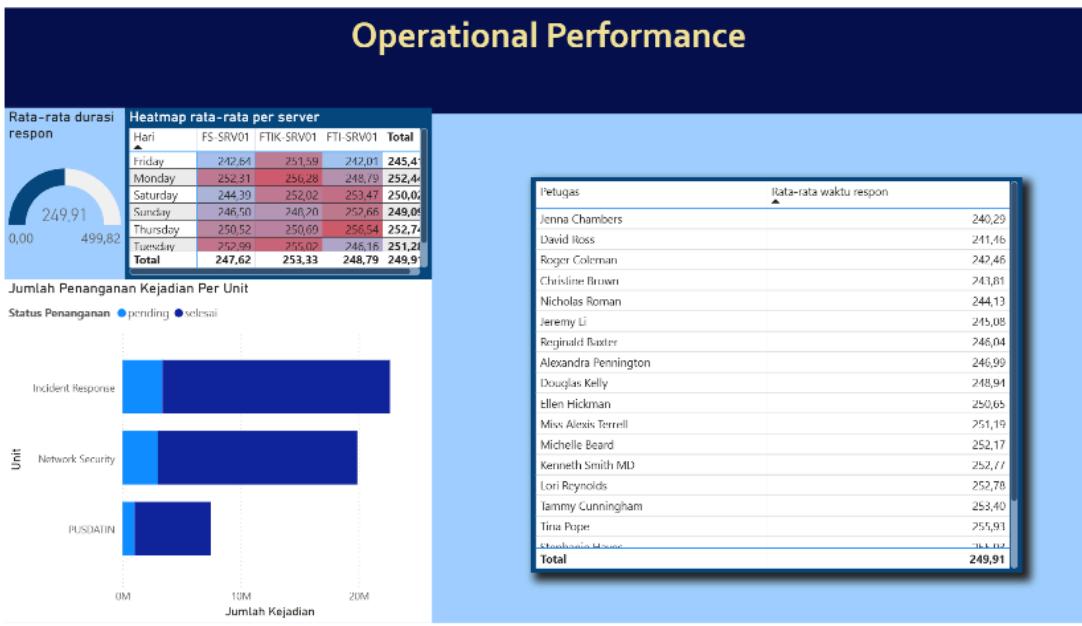
Menampilkan 10 baris teratas

3. Dashboard

a. Executive Summary



b. Operational Performance



c. Impact Analysis

Impact Analysis

