### **CyberSecurity Incident Management System**

The following database schema encompasses tables that manage information related to incidents, vulnerabilities, assets, users, notes associated with incidents and vulnerabilities, as well as user activity logs. Here's a description of this database:

Database Name: IncidentManagementDB

#### **Purpose and Use Case:**

This database is designed to facilitate the management, tracking, and analysis of incidents, vulnerabilities, assets, and user related activities within a system or organization. It allows for comprehensive tracking of incident details, prioritization of vulnerabilities, user authentication, collaboration through notes, and audit tracking of user activities.

#### **Key Functionalities:**

- Incident Management: Tracking and managing incidents from reporting to resolution.
- Asset Management: Organizing and categorizing various system assets.
- User Management: Authentication, role assignment, and user related data storage.
- Communication and Collaboration: Recording notes for incidents and vulnerabilities.
- Audit Trail: Logging user activities for monitoring and compliance purposes.

This database structure supports efficient incident resolution, vulnerability management, asset organization, user administration, and maintains a record of user interactions, contributing to the overall security and operational management within the system or organization.

#### **TABLES**

#### 1. Incidents Table:

IncidentID (Primary Key)

IncidentTypeID (Foreign Key referencing IncidentType table)

Description

Severity

**Status** 

ReportedBy

ReportedDate

ResolvedDate

+		+		++	+
Field	Туре	Null	Key	Default	Extra
IncidentID   Description   Severity   Status	int text varchar(50) varchar(50)	NO   YES   YES   YES	PRI	NULL   NULL   NULL	
ReportedBy   ReportedDate   ResolvedDate   IncidentTypeID	int date date int	YES   YES   YES   YES	MUL	NULL	

### 2.Incident Type:

IncidentType ID (Primary Key)

# TypeName

Field	Type	Null	Key	Default	Extra
IncidentTypeID   TypeName	int   varchar(100)		PRI	NULL NULL	auto_increment   

### 3.Assets Table:

AssetID (Primary Key)

AssetName

AssetTypeID (Foreign Key referencing AssetType table)

Location

Description

Field	Туре	Null	Key	Default	Extra
AssetID AssetName AssetTypeID Location Description	int	NO YES YES YES YES	PRI MUL	NULL NULL NULL NULL NULL	

# 4.AssetType Table:

AssetTypeID (Primary Key)

TypeName

Description (if needed)

+   Field	Туре	Null	Key	Default	Extra
AssetTypeID TypeName Description	int varchar(255) text	NO YES YES	PRI	NULL NULL NULL	

### 5.Users Table:

UserID (Primary Key)

Username

Password (encrypted/hashed)

**Email** 

Role

+   Field +	+   Type 	Null	Key	Default	Extra
UserID Username Password Email Role	int   varchar(100)   varchar(255)   varchar(255)   varchar(50)	NO YES YES YES YES	PRI	NULL NULL NULL NULL NULL	

### 6.IncidentNotes Table:

NoteID (Primary Key)

IncidentID (Foreign Key referencing Incidents table)

NoteContent

CreatedBy

CreatedDate

Field	Type	Null	Key	Default	Extra
NoteID IncidentID NoteContent CreatedBy CreatedDate	int   int   text   int   date	NO YES YES YES YES	PRI   MUL     MUL	NULL NULL NULL NULL NULL	

# 7. User Activity Log Table:

LogID (Primary Key)

UserID (Foreign Key referencing Users table)

ActivityType ID (Foreign Key referencing ActivityType table)

### ActivityDescription

# ActivityDate

Field	Type	Null	Key	Default   Extra
LogID   UserID   ActivityDescription   ActivityDate   ActivityTypeID	int   int   text   datetime   int	NO YES YES YES YES YES	PRI MUL MUL	NULL

### 8.ActivityType Table:

ActivityTypeID (Primary Key)

### TypeName

+   Field	+   Туре	Null	+   Key	Default	Extra
ActivityTypeID   TypeName	int   varchar(100)		PRI 	NULL NULL	auto_increment

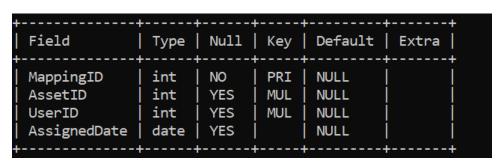
### 9. AssetUserMapping Table:

MappingID (Primary Key)

AssetID (Foreign Key referencing Assets table)

UserID (Foreign Key referencing Users table)

### Assigned Date



**Table Descriptions** 

#### 1.Incidents Table:

- Manages records of various incidents reported within a system.
- Contains details such as incident type, description, severity, status, reporting personnel, reported date, and resolution date if applicable.

#### 2.Incident Type Table:

- Stores different types of incident categories or labels.
- Contains a primary key for the incident type ID and a field for the incident type name.

#### 3.Assets Table:

- Manages data related to various assets within the system infrastructure.
- Includes information such as asset name, asset type ID (linked to the AssetType table), location, and additional descriptions.

### 4.AssetType Table:

- Contains definitions of various asset types or categories.
- Holds primary key AssetTypeID, TypeName, and an optional description field.

#### 5.Users Table:

- Stores information about users interacting with the system.
- Includes details such as unique user IDs, usernames, encrypted/hashed passwords, email addresses, and assigned roles within the system.

#### 6.IncidentNotes Table:

- Associates notes or comments with specific incidents recorded in the Incidents table.
- Facilitates communication by capturing note content, creator details, and creation dates related to incidents.

### 7. User Activity Log Table:

- Logs various activities performed by users within the system.
- Records details like user IDs, activity types (referenced from ActivityType Table), activity descriptions, and timestamps of user actions.

### 8.ActivityType Table:

- Stores different types of user activities or actions.
- Contains a primary key for the activity type ID and a field for the activity type name.

# 9. AssetUserMapping Table:

- Establishes relationships between assets and users.
- Allows assignment of assets to users and tracks the assigned date for effective asset management and allocation.

#### **ER DIAGRAM**

