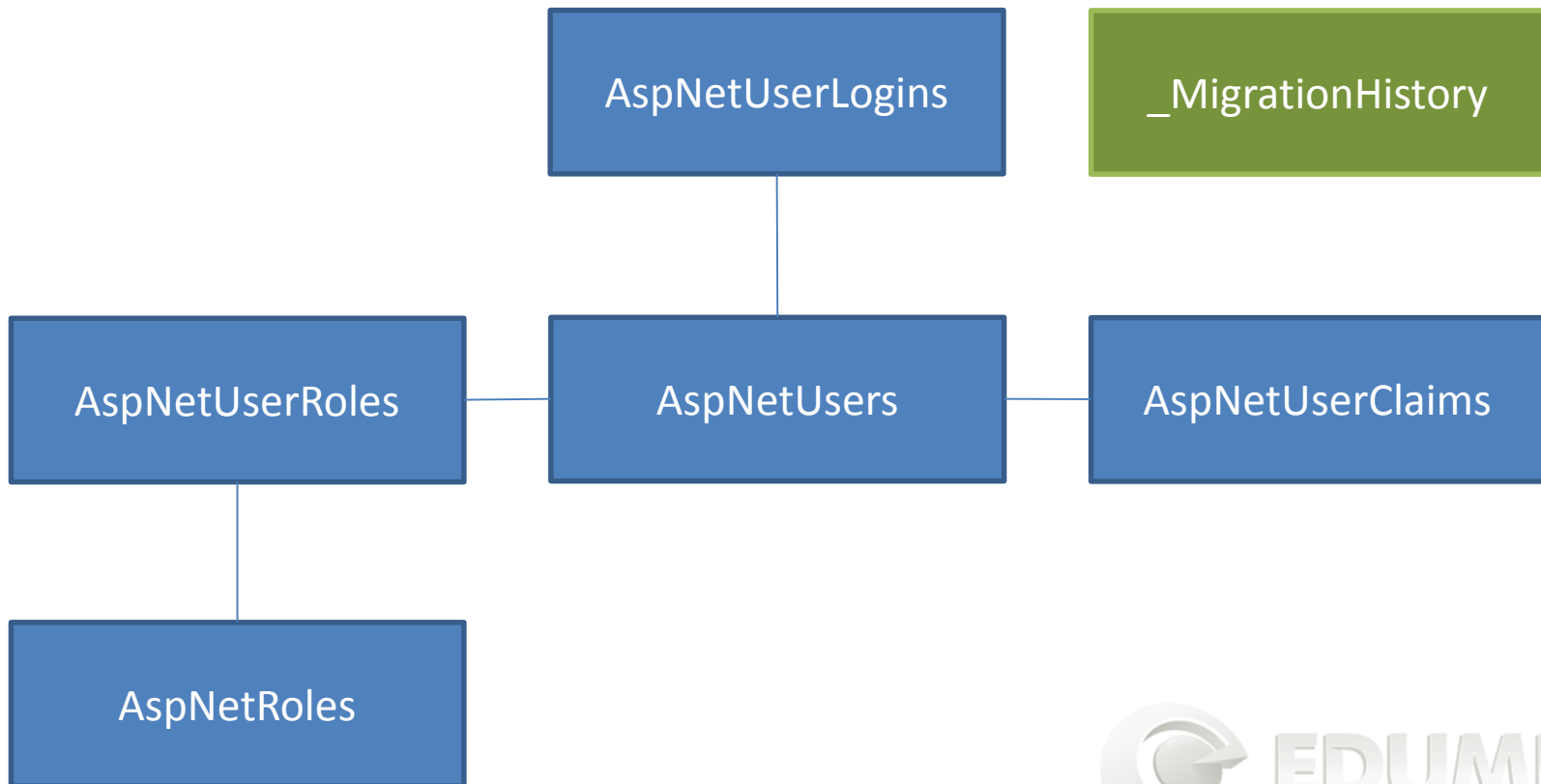# ASP.NET Identity - Database

ASP.NET has its own default database structure for the purpose of handling user information.

| AspNetUserLogins | _MigrationHistory |

| AspNetUserRoles | AspNetUsers | AspNetUserClaims |

| AspNetRoles |

ASP.NET Identity - Database

AspNetUsers

Column	Description	
Id	The user ID (GUID)	Table for storing users and user information
Username	Username	
Email	Users email	
PhoneNumber	Users phone number	
PasswordHash	Hashed password	
SecurityStamp	A unique security token (GUID)	
EmailConfirmed	Confirmation for email	Two Factor Authentication fields
PhoneNumberConfirmed	Confirmation for phone number	
TwoFactorEnabled	Enables email / phone confirmation	
LockOutEnabled	Enables lock out	Account lockout fields
LockOutEndDateUtc	Sets the lock out time	
AccessFailedCount	Number of failed logins attempts	

GUID

- A GUID is a globally unique, randomly generated ,128-bit integer ID

- Extremely low chance of duplication and collision
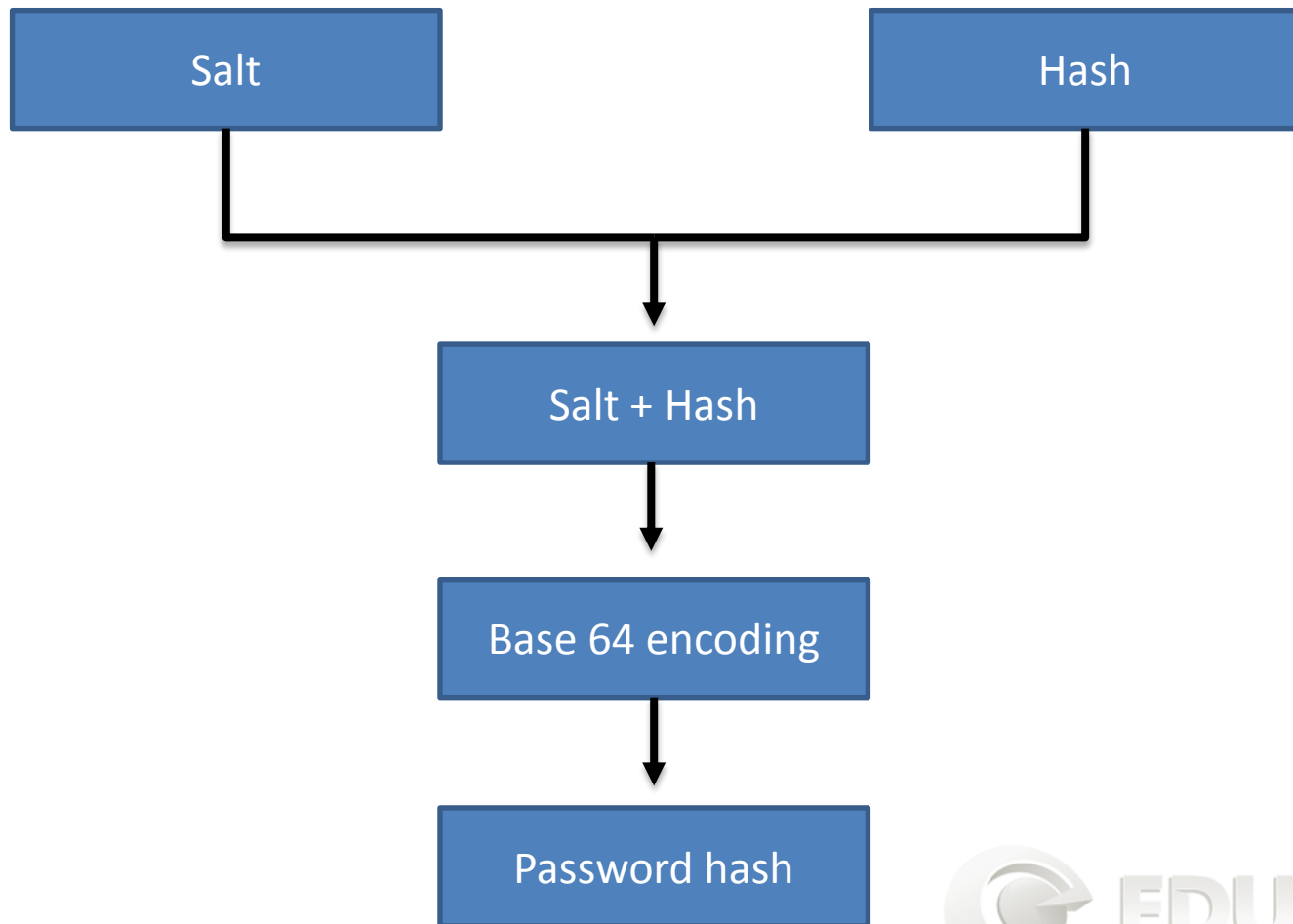
- Total number of possible GUID = 2^{128}

f1454227-eb7f-410a-bd76-b127c4f9e57f

Password Hashing

The password hashing field is generated in the following way:

- Password is hashed
- Hashed using PBFK2 hashing algorithm by default
- Salt contains 16 random bytes
- Salt is added in front of the hashed password
- Base64 encoded

Password Hashing

```
┌─────────────────┐          ┌─────────────────┐
│      Salt       │          │      Hash       │
└─────────────────┘          └─────────────────┘
                  │                    │
                  └────────┬───────────┘
                           ▼
                  ┌─────────────────┐
                  │   Salt + Hash   │
                  └─────────────────┘
                           │
                           ▼
                  ┌─────────────────┐
                  │ Base 64 encoding│
                  └─────────────────┘
                           │
                           ▼
                  ┌─────────────────┐
                  │  Password hash  │
                  └─────────────────┘
```

Password Hashing

Some of the more common hashing algorithms are:

Name	Hash length	Speed	Memory need	Comment
MD5	128 bits	Fast	Low	Not recommended, very weak
SHA-1	160 bits	Fast	Low	Mathematical weakness exists
SHA-2	224,256, 384 or 512 bits	Fast	Low	Not cracked yet.
Bcrypt	448 bits	Slow	Low	
PBKDF2	Custom	Slow	Low	
Scrypt	Custom	Slow	High	Can be made to consume lots of memory

Table for storing roles

AspNetRoles	
Column	**Description**
Id	A generated GUID value
Name	The name of the role

Connects users with roles

AspNetUserRoles	
Column	**Description**
UserId	The Users ID
RoleId	The Roles ID

Table used to store users using external logins

AspNetUserLogins	
Column	**Description**
LoginProvider	The Login Provider (ex. Google, Facebook)
ProviderKey	Provider specific ID for the user
UserId	The users ID

LoginProvider	ProviderKey	UserId
Facebook	10152440515362227	80db0b4f-687d-4fdc-bbb5...
Google	112045923222932727715	ffa0efc9-3d76-4680-8668-6...

Table used to store claims

AspNetUserClaims	
Column	**Description**
Id	An auto incrementing ID
UserId	The users ID
ClaimType	The type of the claim
ClaimValue	The value of the claim

Id	UserId	ClaimType	ClaimValue
1	7120aa6f-f7e2-...	Role	Administator
2	7120aa6f-f7e2-...	Name	Billy
3	7120aa6f-f7e2-...	Country	Sweden

ASP.NET uses the migration feature in entity framework to generate the database tables

Table used to store migration history

_MigrationHistory	
Column	**Description**
MigrationId	ID for migration
ContextKey	Key for database context
Model	
ProductVersion	Entity Framework version