CHAPTER 9

Database Security and Security Patterns

“If you want total security, go to prison. There you’re fed, clothed, given medical care and so on. The only thing lacking . . . is freedom.”

—Dwight D. Eisenhower

There are so many threats to your security that it is essential to remain ever vigilant—without ending up with your server in a bunker of lead wearing a tinfoil hat protecting data by keeping it completely inaccessible to any human eyes. Business needs connectivity to customers, and customers need connectivity to their data. Security is one of the most important tasks when setting up and creating a new application, yet it is often overlooked and dealt with late in the application building process. Whether or not this is acceptable is generally up to your requirements and how your application will be built, but at one point or another, your application team must take the time to get serious about security. Over and over, stories in the news report data being stolen, and the theft is inevitably due to poor security. In the last edition of this book, I used the example of an election official’s stolen laptop in my home city of Nashville, Tennessee; names, addresses, and partial social security numbers were stolen. Since then, there has been a steady stream of such stories, and probably the most high profile has been Sony’s Playstation network getting hacked and being down for months. Hence, if you are the architect of a database system that holds personal and private information, it could be you who becomes jobless with a stain on your career the size of the Mojave desert and possibly quite unemployed if it turns out to be your fault that data leaked out into the hands of some junkie looking for stuff to hock.

Security is on the minds of almost every company today, as evidenced by the constant stream of privacy policies that we continue to come across these days. They’re everywhere and if your company does business with anyone, it likely has one too. Let’s be clear: for far too many organizations, most security is implemented by hoping average users are as observant as Dorothy and her shoes. They have a lot of power if they just were adventurous enough to open a tool like Management Studio and start clicking around (or perhaps clicking their ruby slipper heels together three times.) Of course, fear is a reasonably good motivator for sticking to the marked path, and most average users aren’t too adventurous in the first place (work the help desk for a week, and you will know exactly what I mean). If they were, they’d not only discover how to fix the same problem they had yesterday, but also may just find that they have incredible power to see more than they need to see or to get back home to Kansas in the blink of an eye.
In this chapter, we will be covering the following topics:

- **Database security prerequisites**: We will cover some of the fundamentals that you need to understand before dealing with database-level access.

- **Database securables**: Once you are in the context of a database, you have a lot of built-in control over what users can access. In this section, we will cover what they are.

- **Controlling access to data via T-SQL coded objects**: We will look beyond direct access to data, at how you can restrict access to data in more granular ways using T-SQL procedures, views, and so on.

- **Crossing database lines**: Databases are ideally independent containers, but on occasion, you will need to access data that is not stored within the confines of the database. In this section, we will cover some of the caveats when implementing cross database access.

- **Obfuscating data**: Often, you cannot prevent a user from having access to some data, but you want the program to be able to decode the data only situationally. This is particularly important for personally identifiable data or financial data, so we encrypt the data to keep eyes out except where allowable.

- **Monitoring and auditing**: Turning on a “security camera” and watch what people are doing is sometimes the only real way to verify that you can provide adequate security, and in many cases you will do this and the aforementioned items.

Overall, we will cover a solid sampling of what you will need to secure your data but not the complete security picture, especially if you start to use some of the features of SQL Server that we are not covering in this book (Service Broker to name one). The goal of this chapter will be to shine a light on what is available, demonstrate some of the implementation patterns you may use, and then let you dig in for your exact needs.

I should also note that not everyone will use many, if any, of the guidelines in this chapter in their security implementations. Often, the application layer is left to implement the security alone, showing or hiding functionality from the user. This approach is common, but it can leave gaps in security, especially when you have to give users ad hoc access to the data or you have multiple user interfaces that have to implement different methods of security. My advice is to make use of the permissions in the database server as much as possible. However, having the application layer control security isn’t a tremendous hole in the security of the organization, as long as the passwords used are seriously complex, encrypted, and extremely well guarded and ideally the data is accessed using Windows Authentication from the middle tier.

## Database Access Prerequisites

In this initial section of this chapter, we are going to cover a few prerequisites that we will need for the rest of this chapter on database security. As a programmer, I have generally only been an advisor on how to configure most of the server beyond the confines of the individual database. Setting up the layers of security at the SQL Server instance and Windows Server level is not tremendously difficult, but it is certainly outside of the scope of this book on database design.

As a bit of an introduction to the prerequisites, I am going to cover a few topics to get you started on your way to implementing a secure environment:

- **Guidelines for server security**: In this section, I will cover some of the things you can use to make sure your server is configured to protect against outside harm.

- **Principals and securables**: All security in SQL Server is centered around principals (loosely, logins and users) and securables (stuff that you can limit access to).