MediRoutes Data Security, Data Backup & Disaster Recovery Process

Last Modified December 15th, 2018 by David Reinkensmeyer, Director of Product Development

Overview

This document summarizes Schedule Viewer LLC's ongoing automated data security, data backup and automated disaster recovery measures in place for the MediRoutes Software Platform.

Data Security

Data at Rest (Data Storage)

All MediRoutes data at rest is stored securely in *Microsoft's Azure Cloud* using industry standard encryption, formally known as *TDE (Transparent Data Encryption).*
Read more: https://docs.microsoft.com/en-us/azure/security/azure-security-encryption-atrest and https://docs.microsoft.com/en-us/sql/relational-databases/security/encryption/transparent-data-encryption?view=sql-server-2017

Data in Transit

All MediRoutes data in transit (for example, between the server and the MediRoutes Mobile Driver App) is handled securely via industry standard SSL TLS 1.2 encryption.

Automatic Data Backup

MediRoutes is hosted in *Microsoft's Azure Cloud* using near-real-time geo-redundant databases, formally known as *Azure Backup*.
Read more: https://docs.microsoft.com/en-us/azure/backup/backup-introduction-to-azure-backup

In simple terms, all MediRoutes data is stored in at least two identical databases that are in two separate regions of the United States (Primary = South Central US and Secondary = North Central US, specifically) and the databases automatically sync with each other continuously.
Read more: https://azure.microsoft.com/en-us/global-infrastructure/regions/

Azure Geo-redundant storage (GRS) is designed to provide at least 99.99999999999999% (16 9's) durability of data over a given year by replicating MediRoutes data to a secondary region that is hundreds of miles away from the primary region.
Read more: https://docs.microsoft.com/en-us/azure/storage/common/storage-redundancy-grs

Disaster Recovery Process

In the case of a disaster, MediRoutes is configured through *Azure Traffic Manager* to check the health of MediRoutes resources and route the traffic from the non-healthy resource to the healthy resource, automatically failing over from the primary to our secondary region.
Read more: https://azure.microsoft.com/en-us/services/traffic-manager/

Typically, the primary region is actively handling 99% of all network requests from MediRoutes users. The traffic is directed to the secondary region only when the primary region experiences a service disruption. In that case, all new network requests route to the secondary region.

Since the backup of the database is near instantaneous, both the load balancers have IPs that can be health checked, and the instances are always up and running, this topology provides an option for going in for a low recovery time and failover without any manual intervention.