To move data from an on-premises SQL Server table to Azure Storage as a CSV file using Azure Data Factory (ADF) with ExpressRoute and TLS 1.2, follow these step-by-step instructions:

**Prerequisites:**

1. **Azure Data Factory**: You should have an Azure Data Factory instance set up in your Azure subscription.
2. **Self-hosted Integration Runtime**: Configure a self-hosted integration runtime on a machine in your on-premises network, ensuring it has connectivity to your on-premises SQL Server and Azure services via ExpressRoute with TLS 1.2 enabled.
3. **Permissions**: Ensure you have the necessary permissions and credentials for accessing the on-premises SQL Server and writing to Azure Storage.

**Step-by-Step Guide:**

1. **Create Linked Services**:
   * In your ADF instance, create two linked services: one for your on-premises SQL Server and another for your Azure Storage account.
   * Configure the SQL Server linked service with the connection details, including the server name, database name, username, password, and ensure it uses TLS 1.2.
2. **Create Datasets**:
   * Create two datasets: one for the SQL Server table and another for the Azure Storage container where you want to store the CSV file.
   * Configure the SQL Server dataset to point to the specific table you want to export.
3. **Create a Data Copy Pipeline**:
   * Create a new data pipeline within your ADF instance.
4. **Add Activities to the Pipeline**:
   * Add a "Copy Data" activity to the pipeline. This activity will be used to copy data from the SQL Server table to Azure Storage.
5. **Configure Source and Sink**:
   * In the "Copy Data" activity, configure the source dataset (SQL Server dataset) and the sink dataset (Azure Storage dataset).
   * For the sink, specify that you want to write the data as a CSV file.
6. **Mapping and Transformation (Optional)**:
   * If you need to transform the data or specify which columns to export, define mapping and transformation settings within the "Copy Data" activity.
7. **Set TLS 1.2 for Data Movement**:
   * Ensure that the self-hosted integration runtime used by ADF is configured to use TLS 1.2 for data movement. Check the integration runtime settings to confirm this.
8. **Trigger the Pipeline**:
   * Trigger the pipeline to start the data copy operation.
9. **Monitor and Verify**:
   * Monitor the pipeline's execution to ensure data is successfully copied from the on-premises SQL Server table to Azure Storage as a CSV file.
10. **Validation**:
    * After the pipeline has completed successfully, validate the CSV file in your Azure Storage account to ensure it contains the expected data.

By following these steps, you can move data from your on-premises SQL Server table to Azure Storage as a CSV file using Azure Data Factory with ExpressRoute and TLS 1.2. This process enables secure data transfer between your on-premises environment and Azure while maintaining the necessary security protocols.