**Rule migration checklist**

Before starting the journey to migrate rules from your current SIEM to Azure Sentinel, make sure you do the following preliminary steps to increase your migration success:

|  |  |  |
| --- | --- | --- |
| **No** | **Item** | **Check Box** |
| 1 | **Review all the [Azure Sentinel built-in rules](https://github.com/Azure/Azure-Sentinel/tree/master/Detections)** to identify out-of-the-box rules that can address your use-cases. If there are built-in rules you can use, you’ll need to migrate fewer rules from your current SIEM. |  |
| 2 | **Explore community resources**, such as the SOC Prime [Threat Detection Marketplace](https://my.socprime.com/tdm/) for additional rules you can use instead of migrating your current rules. |  |
| 3 | **Confirm connected data sources** and review [data connection methods](https://docs.microsoft.com/azure/sentinel/connect-data-sources). |  |
| 4 | Identity and prioritize use cases to be migrated These should answer the question - What problems are we trying to solve?  Consider use cases in terms of business priority |  |
| 5 | **Review the detection efficacy of existing rules** before deciding to migrate them into Azure Sentinel. Only migrate those rules that are truly useful. |  |
| 6 | **Review your SOC metrics and consult your SOC team** to identify alerts they routinely ignore without consequence. |  |
| 7 | **Review rules that haven’t triggered any alerts** in the last 6 to 12 months to determine whether they are still relevant. |  |
| 8 | **Eliminate some of the low-level threats or alerts you routinely ignore**. The more you can weed out alerts that you don’t act upon, the more likely the higher-value alerts are to be acted upon. |  |
| 9 | **Define test scenarios and build a test script** to be used for rule validation. |  |