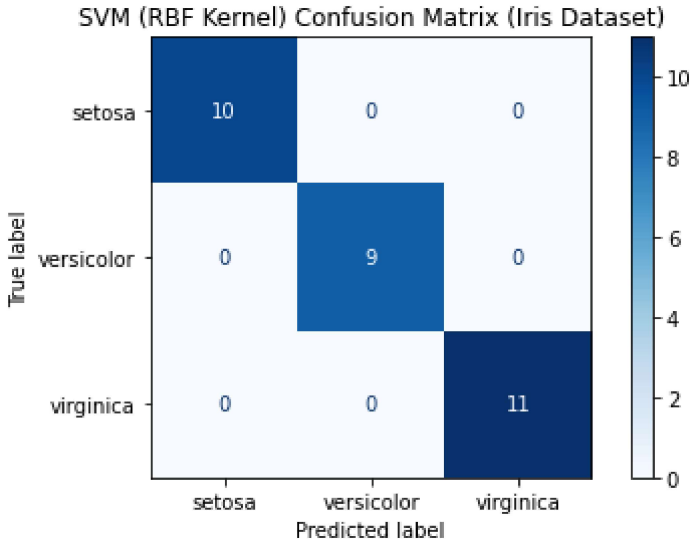
In [6]:

```python
#SVM mit RBF-Kernel auf dem Iris-Datensatz (80:20 Split)

from sklearn import datasets
from sklearn.model_selection import train_test_split
from sklearn.svm import SVC
from sklearn.metrics import confusion_matrix, ConfusionMatrixDisplay, accuracy_score, classification_report
import matplotlib.pyplot as plt

# to check the time of execution, import function time
import time


# Iris-Datensatz laden
iris = datasets.load_iris()
X = iris.data
y = iris.target
class_names = iris.target_names

# Trainings- und Testdaten (80/20)
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# SVM-Modell mit RBF-Kernel
svm_model = SVC(kernel='rbf')
svm_model.fit(X_train, y_train)

# Vorhersage auf Testdaten
y_pred = svm_model.predict(X_test)

# Confusion Matrix
cm = confusion_matrix(y_test, y_pred, labels=[0, 1, 2])
disp = ConfusionMatrixDisplay(confusion_matrix=cm, display_labels=class_names)
disp.plot(cmap='Blues')
plt.title("SVM (RBF Kernel) Confusion Matrix (Iris Dataset)")
plt.tight_layout()
plt.show()

# Accuracy und Classification Report
accuracy = accuracy_score(y_test, y_pred)
print("Accuracy:", accuracy)
print("\nClassification Report:\n")
print(classification_report(y_test, y_pred, target_names=class_names))
```




```
Accuracy: 1.0

Classification Report:

              precision    recall  f1-score   support

      setosa       1.00      1.00      1.00        10
  versicolor       1.00      1.00      1.00         9
   virginica       1.00      1.00      1.00        11

    accuracy                           1.00        30
   macro avg       1.00      1.00      1.00        30
weighted avg       1.00      1.00      1.00        30
```

In [7]:

```
1  # print current date and time
2  print("date & time:",time.strftime("%d.%m.%Y %H:%M:%S"))
3  print ("*** End of Program ***")
4
```

```
date & time: 07.04.2025 11:04:33
*** End of Program ***
```