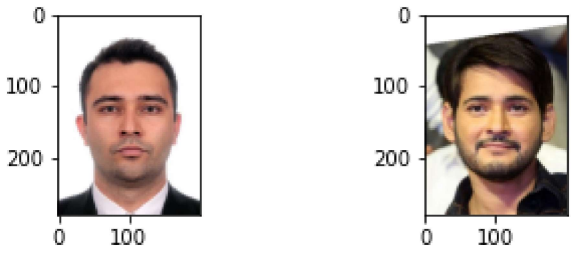
In [17]:
```
from deepface import DeepFace
import cv2
import matplotlib.pyplot as plt
```

In [18]:
```
model_name = "VGG-Face"
distance_metric = "cosine"
```

In [21]:
```
suspect_img = "Target\\DervisEmreAydin.jpg"
betaface_img1 = "BetaFace\\1.jpg"

Suspect_img = cv2.imread(suspect_img)
Betaface_img1 = cv2.imread(betaface_img1)

fig = plt.figure()
ax1 = fig.add_subplot(2,2,1)
ax1.imshow(Suspect_img[:,:,::-1])
ax2 = fig.add_subplot(2,2,2)
ax2.imshow(Betaface_img1[:,:,::-1])

verify1_VGG= DeepFace.verify(suspect_img, betaface_img1,
                          model_name= "VGG-Face",
                          distance_metric= "cosine")

verify1_VGG_list = [verify1_VGG.get('distance'),
                verify1_VGG.get('model'),
                verify1_VGG.get('similarity_metric')]

verify1_Facenet= DeepFace.verify(suspect_img, betaface_img1,
                          model_name= "Facenet",
                          distance_metric= "euclidean_l2")

verify1_Facenet_list = [verify1_Facenet.get('distance'),
                verify1_Facenet.get('model'),
                verify1_Facenet.get('similarity_metric')]

print(verify1_VGG_list)
print(verify1_Facenet_list)
```

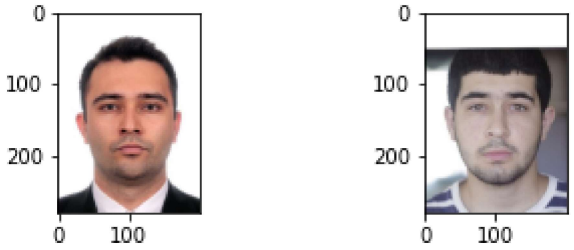
```
[0.30944469366747784, 'VGG-Face', 'cosine']
[0.9104436121403322, 'Facenet', 'euclidean_l2']
```



In [22]:
```
suspect_img = "Target\\DervisEmreAydin.jpg"
betaface_img2 = "BetaFace\\2.jpg"

Suspect_img = cv2.imread(suspect_img)
Betaface_img2 = cv2.imread(betaface_img2)

fig = plt.figure()
ax1 = fig.add_subplot(2,2,1)
ax1.imshow(Suspect_img[:,:,::-1])
```

```
ax2 = fig.add_subplot(2,2,2)
ax2.imshow(Betaface_img2[:,:,::-1])

verify2_VGG= DeepFace.verify(suspect_img, betaface_img2,
                           model_name= "VGG-Face",
                           distance_metric= "cosine")

verify2_VGG_list = [verify2_VGG.get('distance'),
                verify2_VGG.get('model'),
                verify2_VGG.get('similarity_metric')]

verify2_Facenet= DeepFace.verify(suspect_img, betaface_img2,
                           model_name= "Facenet",
                           distance_metric= "euclidean_l2")

verify2_Facenet_list = [verify2_Facenet.get('distance'),
                verify2_Facenet.get('model'),
                verify2_Facenet.get('similarity_metric')]

print(verify2_VGG_list)
print(verify2_Facenet_list)
```

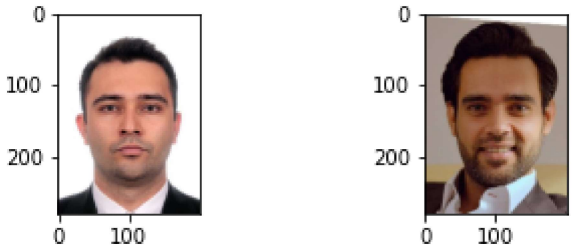
```
[0.33102007977987324, 'VGG-Face', 'cosine']
[1.0300502232076016, 'Facenet', 'euclidean_l2']
```



In [23]:

```
suspect_img = "Target\\DervisEmreAydin.jpg"
betaface_img3 = "BetaFace\\3.jpg"

Suspect_img = cv2.imread(suspect_img)
Betaface_img3 = cv2.imread(betaface_img3)

fig = plt.figure()
ax1 = fig.add_subplot(2,2,1)
ax1.imshow(Suspect_img[:,:,::-1])
ax2 = fig.add_subplot(2,2,2)
ax2.imshow(Betaface_img3[:,:,::-1])

verify3_VGG= DeepFace.verify(suspect_img, betaface_img3,
                           model_name= "VGG-Face",
                           distance_metric= "cosine")

verify3_VGG_list = [verify3_VGG.get('distance'),
                verify3_VGG.get('model'),
                verify3_VGG.get('similarity_metric')]

verify3_Facenet= DeepFace.verify(suspect_img, betaface_img3,
                           model_name= "Facenet",
                           distance_metric= "euclidean_l2")

verify3_Facenet_list = [verify3_Facenet.get('distance'),
                verify3_Facenet.get('model'),
                verify3_Facenet.get('similarity_metric')]

print(verify3_VGG_list)
print(verify3_Facenet_list)
```

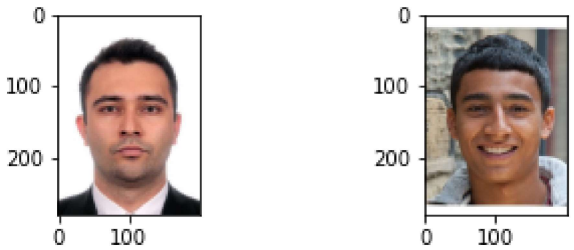
```
[0.23861789867866767, 'VGG-Face', 'cosine']
[1.021729940103321, 'Facenet', 'euclidean_l2']
```



In [24]:

```
suspect_img = "Target\\DervisEmreAydin.jpg"
betaface_img4 = "BetaFace\\4.jpg"

Suspect_img = cv2.imread(suspect_img)
Betaface_img4 = cv2.imread(betaface_img4)

fig = plt.figure()
ax1 = fig.add_subplot(2,2,1)
ax1.imshow(Suspect_img[:,:,::-1])
ax2 = fig.add_subplot(2,2,2)
ax2.imshow(Betaface_img4[:,:,::-1])

verify4_VGG= DeepFace.verify(suspect_img, betaface_img4,
                             model_name= "VGG-Face",
                             distance_metric= "cosine")

verify4_VGG_list = [verify4_VGG.get('distance'),
                    verify4_VGG.get('model'),
                    verify4_VGG.get('similarity_metric')]

verify4_Facenet= DeepFace.verify(suspect_img, betaface_img4,
                             model_name= "Facenet",
                             distance_metric= "euclidean_l2")

verify4_Facenet_list = [verify4_Facenet.get('distance'),
                    verify4_Facenet.get('model'),
                    verify4_Facenet.get('similarity_metric')]

print(verify4_VGG_list)
print(verify4_Facenet_list)
```

```
[0.3520589814705143, 'VGG-Face', 'cosine']
[1.0686986150275701, 'Facenet', 'euclidean_l2']
```

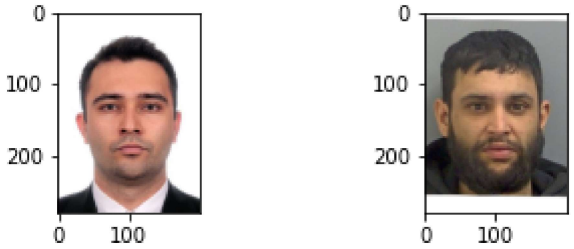


In [25]:

```
suspect_img = "Target\\DervisEmreAydin.jpg"
betaface_img5 = "BetaFace\\5.jpg"

Suspect_img = cv2.imread(suspect_img)
Betaface_img5 = cv2.imread(betaface_img5)

fig = plt.figure()
ax1 = fig.add_subplot(2,2,1)
```

```
ax1.imshow(Suspect_img[:,:,::-1])
ax2 = fig.add_subplot(2,2,2)
ax2.imshow(Betaface_img5[:,:,::-1])

verify5_VGG= DeepFace.verify(suspect_img, betaface_img5,
                          model_name= "VGG-Face",
                          distance_metric= "cosine")

verify5_VGG_list = [verify5_VGG.get('distance'),
                verify5_VGG.get('model'),
                verify5_VGG.get('similarity_metric')]

verify5_Facenet= DeepFace.verify(suspect_img, betaface_img5,
                          model_name= "Facenet",
                          distance_metric= "euclidean_l2")

verify5_Facenet_list = [verify5_Facenet.get('distance'),
                verify5_Facenet.get('model'),
                verify5_Facenet.get('similarity_metric')]

print(verify5_VGG_list)
print(verify5_Facenet_list)
```

```
[0.2422080625186973, 'VGG-Face', 'cosine']
[0.9236730031667729, 'Facenet', 'euclidean_l2']
```

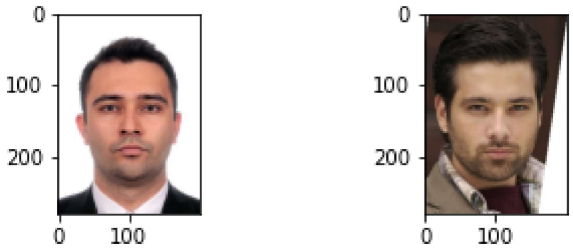


In [26]:

```
suspect_img = "Target\\DervisEmreAydin.jpg"
betaface_img6 = "BetaFace\\6.jpg"

Suspect_img = cv2.imread(suspect_img)
Betaface_img6 = cv2.imread(betaface_img6)

fig = plt.figure()
ax1 = fig.add_subplot(2,2,1)
ax1.imshow(Suspect_img[:,:,::-1])
ax2 = fig.add_subplot(2,2,2)
ax2.imshow(Betaface_img6[:,:,::-1])

verify6_VGG= DeepFace.verify(suspect_img, betaface_img6,
                          model_name= "VGG-Face",
                          distance_metric= "cosine")

verify6_VGG_list = [verify6_VGG.get('distance'),
                verify6_VGG.get('model'),
                verify6_VGG.get('similarity_metric')]

verify6_Facenet= DeepFace.verify(suspect_img, betaface_img6,
                          model_name= "Facenet",
                          distance_metric= "euclidean_l2")

verify6_Facenet_list = [verify6_Facenet.get('distance'),
                verify6_Facenet.get('model'),
                verify6_Facenet.get('similarity_metric')]
```

```
print(verify6_VGG_list)
print(verify6_Facenet_list)
```

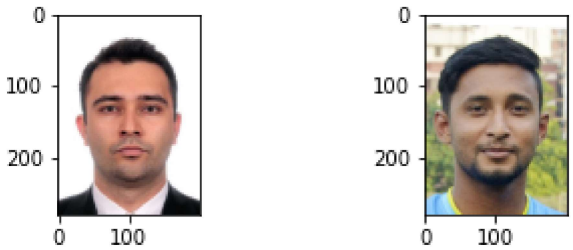
```
[0.21714867479128974, 'VGG-Face', 'cosine']
[1.0231189888364463, 'Facenet', 'euclidean_l2']
```



In [27]:

```
suspect_img = "Target\\DervisEmreAydin.jpg"
betaface_img7 = "BetaFace\\7.jpg"

Suspect_img = cv2.imread(suspect_img)
Betaface_img7 = cv2.imread(betaface_img7)

fig = plt.figure()
ax1 = fig.add_subplot(2,2,1)
ax1.imshow(Suspect_img[:,:,::-1])
ax2 = fig.add_subplot(2,2,2)
ax2.imshow(Betaface_img7[:,:,::-1])

verify7_VGG= DeepFace.verify(suspect_img, betaface_img7,
                          model_name= "VGG-Face",
                          distance_metric= "cosine")

verify7_VGG_list = [verify7_VGG.get('distance'),
                verify7_VGG.get('model'),
                verify7_VGG.get('similarity_metric')]

verify7_Facenet= DeepFace.verify(suspect_img, betaface_img7,
                          model_name= "Facenet",
                          distance_metric= "euclidean_l2")

verify7_Facenet_list = [verify7_Facenet.get('distance'),
                verify7_Facenet.get('model'),
                verify7_Facenet.get('similarity_metric')]

print(verify7_VGG_list)
print(verify7_Facenet_list)
```

```
[0.2644274870354848, 'VGG-Face', 'cosine']
[0.9693951155356866, 'Facenet', 'euclidean_l2']
```

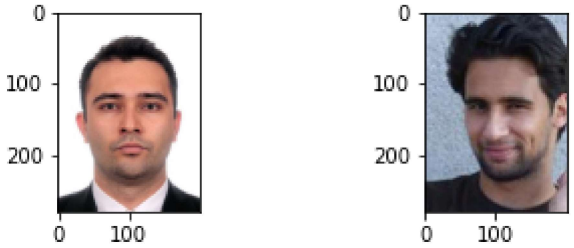


In [28]:

```
suspect_img = "Target\\DervisEmreAydin.jpg"
betaface_img8 = "BetaFace\\8.jpg"

Suspect_img = cv2.imread(suspect_img)
Betaface_img8 = cv2.imread(betaface_img8)
```

```
fig = plt.figure()
ax1 = fig.add_subplot(2,2,1)
ax1.imshow(Suspect_img[:,:,::-1])
ax2 = fig.add_subplot(2,2,2)
ax2.imshow(Betaface_img8[:,:,::-1])

verify8_VGG= DeepFace.verify(suspect_img, betaface_img8,
                           model_name= "VGG-Face",
                           distance_metric= "cosine")

verify8_VGG_list = [verify8_VGG.get('distance'),
                verify8_VGG.get('model'),
                verify8_VGG.get('similarity_metric')]

verify8_Facenet= DeepFace.verify(suspect_img, betaface_img8,
                           model_name= "Facenet",
                           distance_metric= "euclidean_l2")

verify8_Facenet_list = [verify8_Facenet.get('distance'),
                verify8_Facenet.get('model'),
                verify8_Facenet.get('similarity_metric')]

print(verify8_VGG_list)
print(verify8_Facenet_list)
```

```
[0.3717759770242487, 'VGG-Face', 'cosine']
[1.187602314661086, 'Facenet', 'euclidean_l2']
```

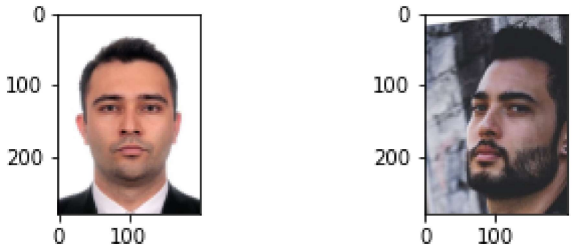


In [29]:

```
suspect_img = "Target\\DervisEmreAydin.jpg"
betaface_img9 = "BetaFace\\9.jpg"

Suspect_img = cv2.imread(suspect_img)
Betaface_img9 = cv2.imread(betaface_img9)

fig = plt.figure()
ax1 = fig.add_subplot(2,2,1)
ax1.imshow(Suspect_img[:,:,::-1])
ax2 = fig.add_subplot(2,2,2)
ax2.imshow(Betaface_img9[:,:,::-1])

verify9_VGG= DeepFace.verify(suspect_img, betaface_img9,
                           model_name= "VGG-Face",
                           distance_metric= "cosine")

verify9_VGG_list = [verify9_VGG.get('distance'),
                verify9_VGG.get('model'),
                verify9_VGG.get('similarity_metric')]

verify9_Facenet= DeepFace.verify(suspect_img, betaface_img9,
                           model_name= "Facenet",
                           distance_metric= "euclidean_l2")

verify9_Facenet_list = [verify9_Facenet.get('distance'),
                verify9_Facenet.get('model'),
                verify9_Facenet.get('similarity_metric')]
```

```
print(verify9_VGG_list)
print(verify9_Facenet_list)
```

```
[0.44614565943766704, 'VGG-Face', 'cosine']
[1.1519312202558205, 'Facenet', 'euclidean_l2']
```



In [30]:

```
suspect_img = "Target\\DervisEmreAydin.jpg"
betaface_img10 = "BetaFace\\10.jpg"

Suspect_img = cv2.imread(suspect_img)
Betaface_img10 = cv2.imread(betaface_img10)

fig = plt.figure()
ax1 = fig.add_subplot(2,2,1)
ax1.imshow(Suspect_img[:,:,::-1])
ax2 = fig.add_subplot(2,2,2)
ax2.imshow(Betaface_img10[:,:,::-1])

verify10_VGG= DeepFace.verify(suspect_img, betaface_img10,
                          model_name= "VGG-Face",
                          distance_metric= "cosine")

verify10_VGG_list = [verify10_VGG.get('distance'),
                verify10_VGG.get('model'),
                verify10_VGG.get('similarity_metric')]

verify10_Facenet= DeepFace.verify(suspect_img, betaface_img10,
                          model_name= "Facenet",
                          distance_metric= "euclidean_l2")

verify10_Facenet_list = [verify10_Facenet.get('distance'),
                verify10_Facenet.get('model'),
                verify10_Facenet.get('similarity_metric')]

print(verify10_VGG_list)
print(verify10_Facenet_list)
```

```
[0.44481162283828046, 'VGG-Face', 'cosine']
[1.165845047826171, 'Facenet', 'euclidean_l2']
```