In [1]:
```
from django.contrib.auth.models import User
from django.contrib.auth.password_validation import validate_password
import plotly.graph_objects as go
```

In [2]:
```
def show_graph(total, protect):
    fig = go.Figure(go.Indicator(
        mode = "gauge+number",
        value = protect,
        domain = {'x': [0, 1], 'y': [0, 1]},
        title = {'text': "Protection fee"},
        gauge = {'axis': {'range': [None, total]}},
    ))

    fig.update_layout(autosize=False, width=500, height=500)

    fig.show()
```

Test from the 10.000 most common passwords

In [3]:
```
total, protect, insecure = 0, 0, 0
file_passwords = open('10000_passwords.txt', 'r')

for i, line in enumerate(file_passwords):
    password = line.replace('\n', '')
    total += 1
    try:
        validate_password(password)
        insecure += 1
    except Exception as e:
        #print(i, e)
        protect += 1
file_passwords.close()
result = "Protected: {} | Insecure {} | Protection fee: {}%".format(protect, insecure, protect/total
*100 )
show_graph(total, protect)
print(result)
```


```
Protected: 9553 | Insecure 447 | Protection fee: 95.53%
```

Test from the 1.000.000 most common passwords

In [5]:
```
total, protect, insecure = 0, 0, 0
file_passwords = open('1000000_passwords.txt', 'r')

for i, line in enumerate(file_passwords):
    password = line.replace('\n', '')
    total += 1
    try:
        validate_password(password)
        insecure += 1
    except Exception as e:
        #print(i, e)
        protect += 1
file_passwords.close()
result = "Protected: {} | Insecure {} | Protection fee: {:.2f}%".format(protect, insecure, protect/t
otal*100 )
show_graph(total, protect)
print(result)
```


```
Protected: 573996 | Insecure 426003 | Protection fee: 57.40%
```

In [ ]: