VIRTCRIME Training: BlockSci Demo Part II (Live)

Notwendige Python-Module:

In [1]:
```
import blocksci
import matplotlib.pyplot as plt
import matplotlib.ticker
import collections
import pandas as pd
import numpy as np
from datetime import datetime
%matplotlib notebook
```

```
Warning: You only have 6GB of free disk space left. Running out of d
isk space may crash the parser and corrupt the BlockSci data files.
```

In [2]:
```
# btc = [block 1, block 2, block 3, ...]
btc = blocksci.Blockchain("/mnt/data/blocksci/bitcoin/595303-root-v
0.6-0e6e863/config.json")
```

Einfache Analyse mit Diagramm: Transaktionen pro Block

In [3]:
```
txesPerBlock = []
```

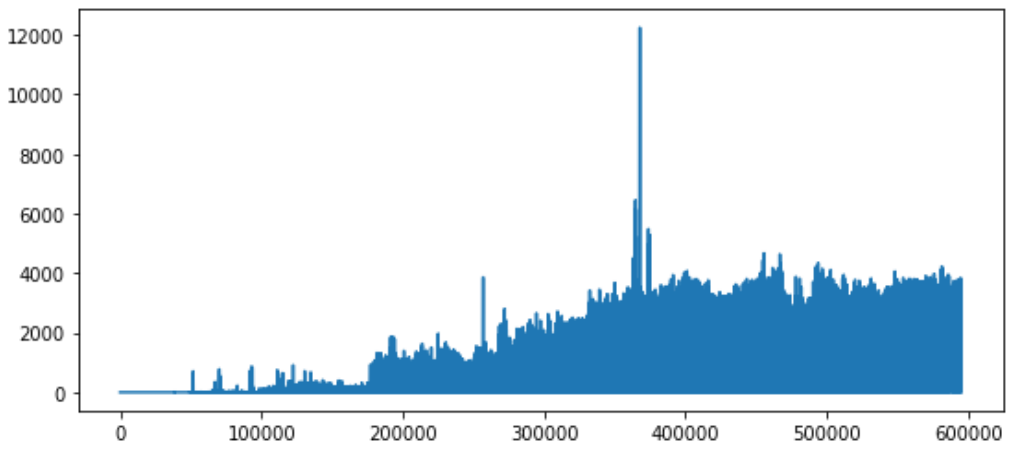
In [4]:
```
for block in btc:
    txesPerBlock.append(len(block))
```

In [5]:
```
txesPerBlock[-10:]
```

Out[5]:
```
[1622, 1318, 1162, 1799, 2854, 1207, 1934, 2861, 2729, 3144]
```

In [6]:
```
plt.figure(figsize=(9,4))
plt.plot(txesPerBlock)
```

Out[6]:
```
[<matplotlib.lines.Line2D at 0x7f81ac78d898>]
```



Transaktionsgebühren von einem Block

In [7]:
```
example_block_height = 450000
df = pd.DataFrame(btc[example_block_height].txes.fee_per_byte(), co
lumns=["Satoshis per byte"])
ax = df.reset_index().plot.scatter(x="index", y="Satoshis per byt
e",figsize=(9,4))
ax.set_ylim(0)
ax.set_xlim(0)
plt.show()
```