

## Abgabe PHYSEC 4

### Bit Error Rate

```
def ber(X, Y):  
    hw = 0.0  
    for x,y in len(X):  
        hw += x^y  
  
    BER = hw/len(X) # len(X) = len(Y)  
  
    if 0 <= BER <= 0.5:  
        return BER  
    else:  
        return 1 - BER
```

## Implementierung Quantisierer

### Implementierung Jana Multibit

```
def JanaMultibit(X):  
    #(i) determine the Range of RSS measurements from the minimum and the maximum  
    measured RSS values  
  
    x_min = min(X)  
    x_max = max(X)  
  
    #(ii) find N, the number of bits that can be extracted per measurement  
    # N is already implemented  
    N = number_of_bits  
  
    #(iii) divide the Range into m = 2^n equal sized intervals  
    M = 2**N # M intervals, len(Range) = M+1  
    interval = abs((max(X)-min(X))/float(M)) # interval size  
  
    Range = [] # intervals  
  
    if interval == 0: #min=max  
        interval = 1  
  
    while x_min <= x_max: # fill in  
        Range.append(x_min)  
        x_min += interval  
  
    #(iv) choose an N bit assignment for each of the M intervals (for example  
    use the Gray code sequence)  
    bit_assignment = utils.gray_code(N)  
  
    #(v) for each RSS measurement, extract N bits depending on the interval in  
    which the RSS measurement lies.  
    quantized=[]  
    for x in X:  
        for j in xrange(len(Range) - 1):  
            if Range[j] <= x <= Range[j+1]: # including max and min values  
                quantized.extend(bit_assignment[j])  
                break  
        if len(Range) == 1: # min=max - only one value  
            quantized.extend(bit_assignment[0])  
    return quantized
```

# Attack Trees

Modifizierbare *draw.io*-Datei hier downloaden zur näheren Inspizierung.

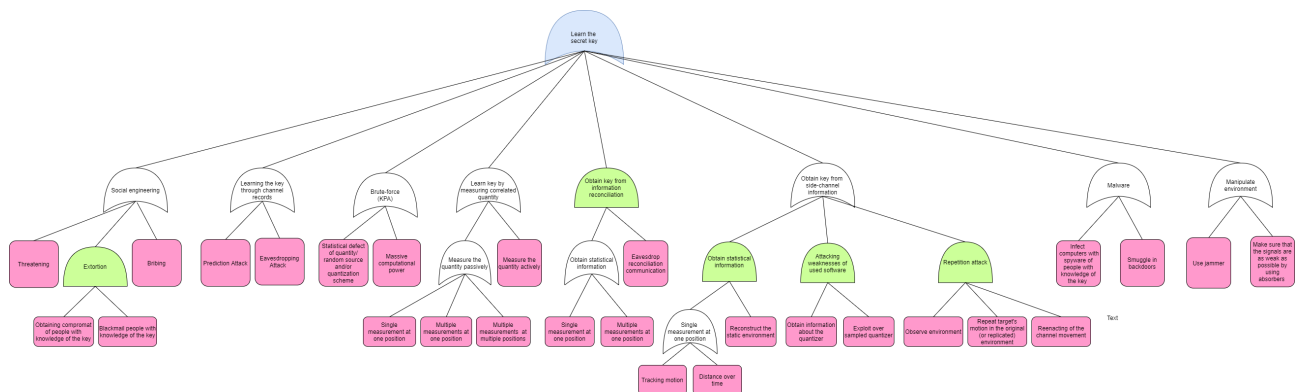


Abbildung 1: Attack tree

Weitere Angriffsszenarien neben

- *Repetition Attack*
- *Prediction Attack*
- *Eavesdropping Attack*
- *Angriff auf den Quantisierer*

sind unter anderem *social engineering* Attacken wie

- *Bestechen*
- *Drohen*
- *Erpressen mithilfe von Kompromat*

oder *Schadsoftwareattacken* wie

- *Spyware*
- *Backdoors*

sowie *Manipulation des Kanals* wie

- *Jammen*
- *Signalabsorber*
- *Attacken wie das Alternieren zwischen Umklammern und Loslassen der Antenne*

## Angriffe

## Reading Assignment

### Nennen Sie 5 verschiedene Quellen für Entropie

#### Physikalische Phänomene

- Radioaktiver Zerfall
- Transmission von Photonen
- Thermisches Rauschen (Johnson-Nyquist-Rauschen)
- Geräusche der Atmosphäre
- Jitter

#### Mensch-Gerät Interaktion

- Dauer von Tastendrücker und andere Zeitmessungen
- Tasten- und Mausbewegungen
- Bewegungs- und Beschleunigungssensoren

### Wie viele Samples aus einem Smartphone Accelerator benötigen wir um einen 128 bit key zu erstellen?

„Since the recommended security level for almost random bits is  $\epsilon = 2^{-80}$ . According to the heuristic (1) we need roughly  $128/0.125 = 1024$  samples to extract a 128-bit key. However taking into account the true error in our Theorem 1 we see that we need at least  $n \approx 2214$  bits!“

Laut dem Paper, welches sie auf Theoreme stützt, braucht man mindestens 1024 Samples. Hinzu kommt allerdings, dass empfohlen wird, ein  $n$  mit mehr als 2214 Bitlänge zu wählen.

### Wie groß ist der Unterschied an tatsächlich extrahierbarer Entropie und Shannon Entropie?

**Corollary 1 (A Significant Entropy Loss in the AEP Heuristic Estimate).** *In the above setting, the gap between the Shannon entropy and the number of extractable bits  $\epsilon$ -close to uniform equals at least  $\Theta(\sqrt{\log(1/\epsilon)kn})$ . In particular, for the recommended security level ( $\epsilon = 2^{-80}$ ) we obtain the loss of  $kn - N \approx \sqrt{80kn}$  bits, no matter what an extractor we use.*

Abbildung 2: Auszug aus dem Paper

„In information theory most widely used is Shannon entropy, which quantifies the encoding length of a given distribution. In turn, cryptographers use the more conservative notion called min-entropy, which quantifies unpredictability. In general, there is a large gap between these two measures: the min-entropy of an  $n$ -bit string might be only  $O(1)$  whereas its Shannon entropy as big as  $\Omega(n)$ .“

Die min-Entropie eines  $n$ -bit langen stringes kann konstant sein, wobei gleichzeitig seine Shannon Entropie linear oder größer zu  $n$  sein kann.

Von den 2214 bits können nur etwa 128 bits mit nahezu zufällig gleichverteilter Qualität extrahiert werden. Das ist ein Verhältnis von  $\frac{128}{2214} \approx 0.0578$  oder anders gesagt: Der Schlüssel hat hier nur die Länge von 5,78% von der Bitlänge des Samples.