

M183 Applikationssicherheit Implementieren

Tutorial zum Lab Encryption One Time Pad

Version 1	12.12.2017	Jürg Nietlispach
-----------	------------	------------------

Contents

Idee	3
Herangehensweise Cryptoanalysis	3

Idee

In diesem Lab soll eine .Javascript Applikation erstellt werden, Texte mit dem Prinzip One Time Pad verschlüsselt (und wieder entschlüsselt).

Herangehensweise Cryptoanalysis

1. Index.html File erstellen
2. GUI-Elemente erstellen
3. Eventhandling und Routinen für Verschlüsselung und Entschlüsselung erstellen

Leeres index.html File erstellen mit folgendem Header-Informationen

```
<!doctype html>
<html class="no-js" lang="">
<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css" />

<body>
  <div class="container">
```

Es müssen nun Elemente für den Verschlüsselungsteil erstellt werden:

```
<body>
  <div class="container">
    <div class="row">
      <div class="col-md-12">
        <h1>One Time Pad</h1>
      </div>
    </div>

    <div class="row">
      <div class="col-md-6">
        <br>
        <legend>Plaintext</legend>
        <textarea id="plaintext" class="form-control"></textarea><br>
        <button class="btn btn-primary" id="analyze_plaintext">Analyze Plaintext</button> <br><br>
        <button class="btn btn-primary" id="encrypt">encrypt</button>
      </div>

      <div class="col-md-6">
        <br>
        <legend>Histogram Plaintext</legend>
        <div id="chart_plaintext"></div>
      </div>
    </div>

    <div class="row">
      <div class="col-md-12"><br><br></div>
    </div>
```

Ebenfalls für den Entschlüsselungsteil

div>

Wurden alle GUI Elemente korrekt hinzugefügt, sieht das dann so aus im Browser.

One Time Pad

Plaintext	Histogram Plaintext
<div><div></div><div>Analyze Plaintext</div><div>encrypt</div></div>	
Ciphertext	Histogram Ciphertext
<div>Generated One-Time-Pad:</div> <div><div></div><div>Analyze Ciphertext</div><div>decrypt using key <div></div></div></div>	

Nun müssen die Events registriert werden und die Entschlüsselungs- und Verschlüsselungsroutinen erstellt werden:

```

document.getElementById("analyze_plaintext").addEventListener("click", function (e) {

    e.preventDefault();
    e.stopPropagation();

    var text = document.getElementById("plaintext").value;
    var histogram = createHistogramValues(text);

    document.getElementById("chart_plaintext").innerHTML = createHTMLTable(histogram);

});

document.getElementById("encrypt").addEventListener("click", function (e) {

    e.preventDefault();
    e.stopPropagation();

    var plaintext = document.getElementById("plaintext").value;
    var encrypted_values = encryptOneTimePad(plaintext);
    document.getElementById("ciphertext").innerHTML = encrypted_values.ciphertext;
    document.getElementById("key_onetime_pad").innerHTML = encrypted_values.key;

});

```

Die beiden Histogramm-Funktionen sind wieder dieselben:

```

function createHistogramValues(text)
{
    var histogram_prepare = [];

    for (var i = 0, len = text.length; i < len; i++) {

        var letter = text[i];
        if (letter.match(/[a-z]/i))
        {
            histogram_prepare[letter] = (histogram_prepare[letter] || 0) + 1;
        }
    }

    histogram = histogram_prepare.sort(function(a, b) {
        a = a[1];
        b = b[1];

        return a < b ? -1 : (a > b ? 1 : 0);
    });

    return histogram;
}

function createHTMLTable(histogram)
{
    var html = "<table>";
    for (var key in histogram) {
        html += "<tr>";
        html += "<td>" + key + ": </td><td>&nbsp;</td>";
        html += "<td>" + histogram[key] + "</td>";
        html += "</tr>";
    }
    html += "</table>";
    return html;
}

```

Verschlüsselung und Entschlüsselung mit dem One Time Pad:

```

function encryptOneTimePad(input) {
    // generate key
    var key = '';

    //-- Generate secret key with same length as message --
    for(var k = 0; k < input.length; k++)
    {
        var c = input[k].charCodeAt(0);

        if ((c >= 65) && (c <= 90))
        {
            key += String.fromCharCode(Math.floor(Math.random() * 26) + 65);
        }
        else if ((c >= 97) && (c <= 122))
        {
            key += String.fromCharCode(Math.floor(Math.random() * 26) + 97);
        }
    }

    var output = "";
    for (var i = 0; i < input.length; i++)
    {
        var char = input[i];

        if (char.match(/[a-z]/i))
        {
            var c = parseInt(input.charCodeAt(i));
            var key_char_shift = parseInt(key.charCodeAt(i));

            if ((c >= 65) && (c <= 90)) // uppercase
            {
                output += String.fromCharCode((c - 65 + key_char_shift - 65) % 26 + 65);
            }
            else if ((c >= 97) && (c <= 122)) // lowercase
            {
                output += String.fromCharCode((c - 97 + key_char_shift - 97) % 26 + 97);
            }
        }
    }

    return {"ciphertext" : output, "key" : key };
}

```



```

function decryptOneTimePad(input, key) {
    var output = "";
    for (var i = 0; i < input.length; i++)
    {
        var char = input[i];

        if (char.match(/[a-z]/i))
        {
            var cr = parseInt(input.charCodeAt(i));

            if ((cr >= 65) && (cr <= 90)) // uppercase
            {
                var c = parseInt(cr - 65);
                var key_char_shift = parseInt(key[i].charCodeAt(0) - 65);

                var new_char_position = (c - key_char_shift) % 26 ;
                if(new_char_position < 0)
                    new_char_position += 26;

                output += String.fromCharCode(new_char_position + 65);
            }
            else if ((cr >= 97) && (cr <= 122)) // lowercase
            {
                var c = parseInt(cr - 97);
                var key_char_shift = parseInt(key[i].charCodeAt(0) - 97);

                var new_char_position = (c - key_char_shift) % 26 ;
                if(new_char_position < 0)
                    new_char_position += 26;

                output += String.fromCharCode(new_char_position + 97);
            }
        }
    }

    return output;
}

```

Und hier noch die zugehörigen Events, innerhalb welchen die obigen Routinen angestossen werden:

```

document.getElementById("analyze_ciphertext").addEventListener("click", function (e) {

    e.preventDefault();
    e.stopPropagation();

    var text = document.getElementById("ciphertext").value;
    var histogram = createHistogramValues(text);

    document.getElementById("chart_ciphertext").innerHTML = createHTMLTable(histogram);

});

document.getElementById("decrypt").addEventListener("click", function (e) {

    e.preventDefault();
    e.stopPropagation();

    var plaintext = document.getElementById("ciphertext").value;
    var key = document.getElementById("decryptkey").value;
    document.getElementById("ciphertext").innerHTML = decryptOneTimePad(plaintext, key);

});

```

Für ExpertInnen: GUI Erweitern, dass je Key-Spalte die Frequenzanalyse angezeigt wird. Sollte keine Regelmässigkeiten aufzeigen (-> Pseudo-Random-Generator)