UNAMUR INFOSEC

CRYPTSTRING

A web application for the visualisation of cryptographic transformations of strings.

Programmer's Guide

April 2021

Antoine Mouchet

Contents

1 Introduction

1.1 General purpose

CRYPTSTRING is a web application (available here) allowing the user to apply transformations to a string of characters. It offers the user the possibility to combine those operations in a flexible sequence. Cryptstring helps the user to visualise and to analyse the effects of cryptographic primitives on their given string.

1.2 System requirements

The website should run correctly on the latest version of any widespread web browser except Opera Mini and Internet Explorer.

IE	Edge	Firefox	Chrome	Safari	Opera	Safari on iOS	Opera Mini	Android Browser	Opera Mobile	Chrome for Android	Firefox for Android	UC Browser for Android	Samsung Internet	QQ Browser	Baidu Browser	KaiOS Browser
	12-13			3.1-10		3.2-10.2										
	14	2-51	4-54	10.1	10-41	10.3							4-5.4			
6-10	15-88	52-85	55-88	11-13.1	42-72	11-13.7		2.1-4.4.4	12-12.1				6.2-12.0			
11	89	86	89	14	73	14.5	all	89	62	89	86	12.12	13.0	10.4	7.12	2.5
		87-88	90-92	TP												

Green: compatible, purple: incompatible

2 Structure

The project is divided into 3 files:

1. base.html: HTML code describing the structure of the page and its basic components.

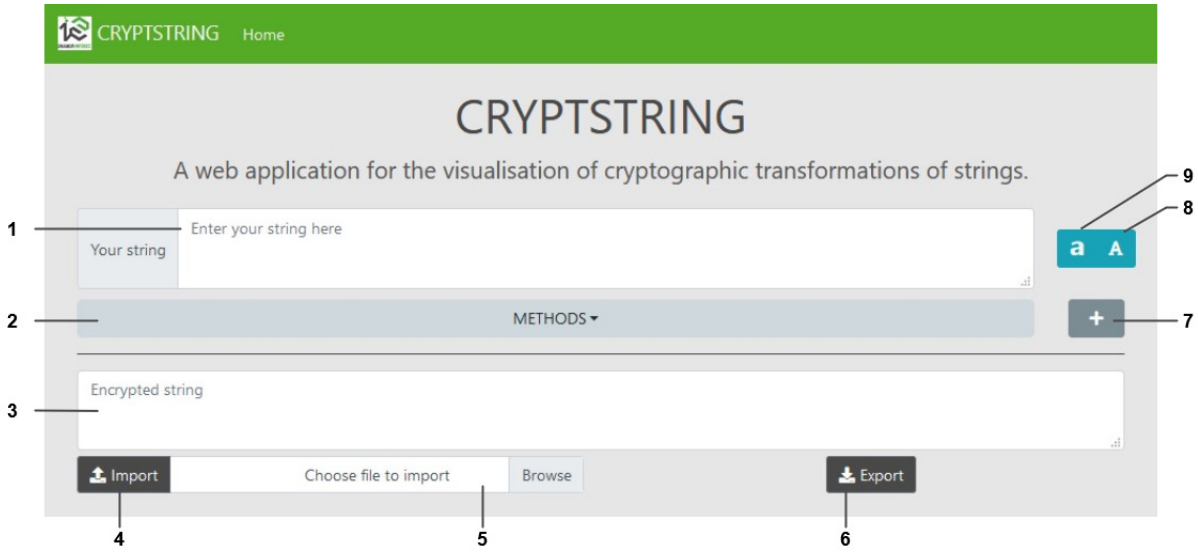
2. base.css: The CSS associated with the homonym HTML file, here for customization purposes.

3. scripts.js: The file containing all functions used in the application as well as intermediate functions and objects used by the app. The methods are defined and stored here.

There is no back end as the application was initially designed to be client side only.

3 The Interface

The interface was designed using components from Bootstrap v4 and icons from Font Awesome.



1. Input field for the string
2. Method selection menu // Drop-down *METHODS* menu
3. Result display field
4. Import button
5. File selection button
6. Export button
7. Add method button
8. Uppercase button
9. Lowercase button

The page can be decomposed into 5 zones (from top to bottom):

- The navigation bar (with only one page *Home* at the moment).
- The title and the description of the application.
- The input field with the buttons to modify the string.
- The method zone with the *METHODS* menu.
- The output zone with the *Import* and *Export* buttons.

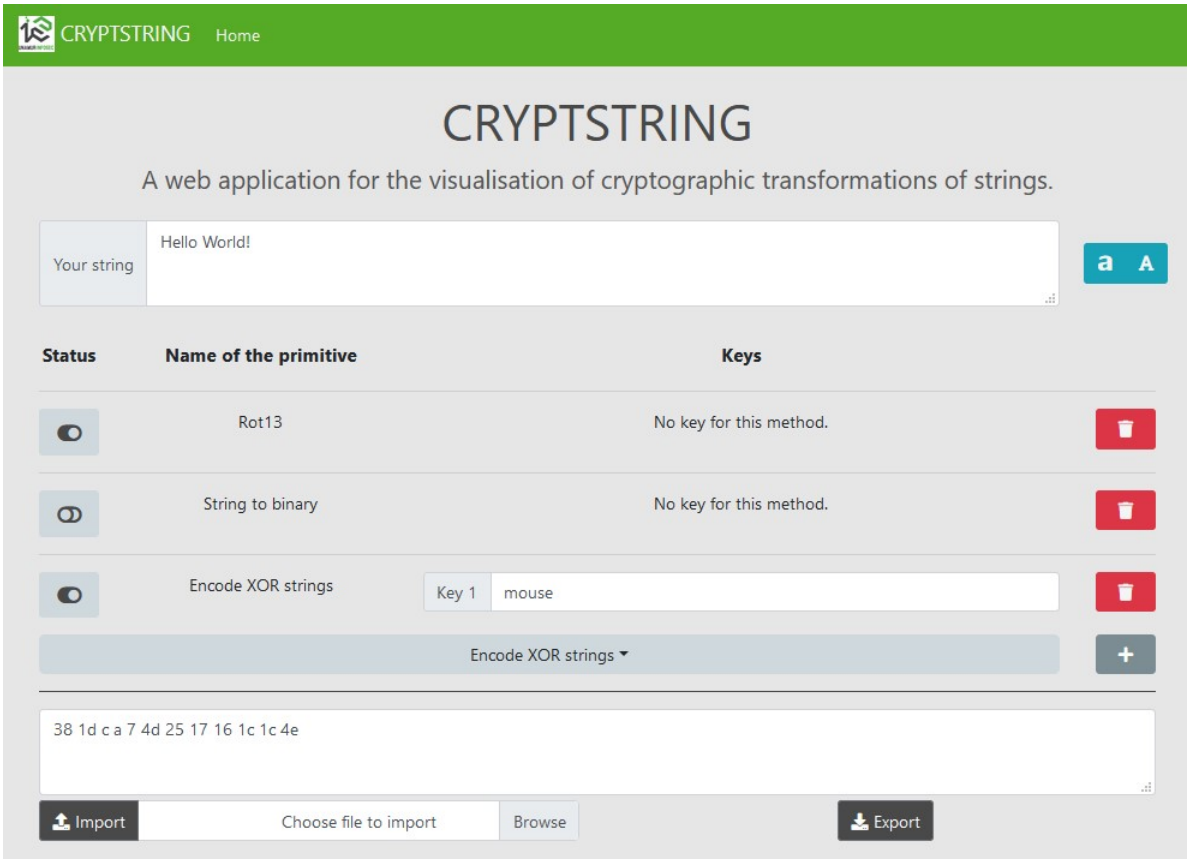
3.1 The Navigation Bar

The navigation bar should link to every page of the application as *href* component and should be present on every page.

3.2 The Input Field & Buttons

The column contains the buttons to modify the string, each button calls a JavaScript function that modifies the string. Any button can be added with its own function.

3.3 The Method Zone



Each method as a field, in its own column describing, its status, its name, a field for its keys (when it is pertinent) and a field with the button to delete the method.

3.4 Methods Menu

The menu contains all available methods. It is built when the browser loads the page using the *populateMethodsMenu* method in *scripts.js*.

4 Data Types

4.1 CryptoMethod

Each transformation available in Cryptstring is represented by a CryptoMethod Object. It has 5 properties:

1. Name of the method
2. Number of keys used by the method
3. List of keys
4. Method associated to the transformation
5. Status of the method

To retrieve the values associated to those values, getters are available. The status can be changed via the *changeStatus* method and a key can be added using the *addKey* method.

4.2 Existing Method

All the existing methods are hardcoded in the *existingMethods* array in the *populateMethods* function. An array is easily traversable and accessible. It also possesses pre-existing methods making its manipulation easier.

4.3 Methods Used

The methods used when adding a method to the list of transformations are stored in the *methodsUsed* array. When adding a new method to this array (via the *addMethod* function), a new CryptoMethod based on the model of the existing method with the selected name is created. The objective is to avoid sharing any reference.

5 Deployment

5.1 Deploy locally

Cryptstring can be used locally very easily.

- Get the project at https://github.com/antoinemouchet/CRYPTSTRING
- Open the *base.html* file.
- Have fun.

5.2 Deploy on Apache Web Server

Because Cryptstring does not need a back-end, it can be deployed easily on any Apache Web Server.

- Get the project at https://github.com/antoinemouchet/CRYPTSTRING
- Connect to your server using a FTP connection.
- Copy the *code* folder from your local computer to the distant server.
- The site should now be available at "[base URL]/code/base.html" .

For more information regarding the upload of files on a web server, consult this MDN Web Docs article.

6 Specifications of functions and objects

A documentation of the code was generated using JSDoc. It is available as an attached file. Just open the *global.html* file from the *JSDoc* folder in your web browser.

7 License

The project is distributed under the Apache License, Version 2.0 and is available at `https://github.com/antoinemouchet/CRYPTSTRING`

8 Future Extension

- More transformations (DES, Vigenère, MD5...).
- More input buttons to change the base string (formatting).
- Networking functionalities (packet analysis)

9 Contact

For any question or suggestion, feel free to send us an email to: amouchet.projects@gmail.com.

10 End Notice

This project is sponsored by Jean-Noel Colin, a member of the InfoSec research team.