

History and relevancy of Enigma machine encryption to present day wireless security issues

Robin Snyder
Savannah State University
126 Jordan, P. O. Box 20359
Savannah, GA 31404
(912) 356-2716
snyderr@savstate.edu
<http://www.RobinSnyder.com>

Abstract

A century ago, the wireless revolution involving radio signals created security problems (at a national level) that are similar to many wireless security problems faced today (at institutional and personal levels). One historical result was the Enigma Machine used by Germany in the second world war. This paper (and talk) will discuss (and present) an overview of this history in general and the Enigma machine in particular. The author has written an Enigma simulator used to generate problems for students. The students then use realistic online Enigma machine software to decrypt the messages. A discussion of current day wireless security involving SID broadcasting, MAC address filtering, WEP, WPA, TKIP, etc., will be covered as an extension of the historical context of the Enigma machine.

Introduction

Secret communication via codes, ciphers, etc., have have existed since ancient times. Some references on the history of codes and ciphers, including the Enigma Machine, are [2], [3], and the classic [1]. Much of the material in this paper is from the author's previous work in this area ([5], [4]).

With the advent of wireless communication, from Marconi's work in the late 19th century, naval commanders in the home country were able to command and control warships throughout the world. Soon, the enemy was always listening so that encryption techniques were continually refined and further developed.

On the wireless networks of today, hackers are often listening. Recently, breaking the standard WEP (Wireless Equivalency Privacy) method has become a game for hackers breaking into Wi-Fi networks. The time to break such encryption is now under a minute. The WPA (Wi-Fi Protected Access) method is much more secure, but as it is not the out-of-the-box default, many users do not use WPA encryption - with a pre-shared (and secure) key.

At the start of the 20th century, there was a desperate need for secure wireless communication. Various methods were developed, refined, and used. Much of the early security relied on code books whereby recognizable words were looked up in a dictionary to find their meaning. Such codes were unwieldy, not that hard to break, etc. A general purpose cipher system was needed.

Between World War I and World War II, the news of the breaking of the German codes during the war by the British and the news of the breaking of the Japanese codes (after the war) by the

2007 ASCUE Proceedings

Americans motivated both Germany and Japan to search out more secure, cipher-based, methods of encrypted communication. The technology developed by the Germans, and the breaking of that technology by the Allies provided the basis for the modern general purpose programmable computer.

Germany adopted a version of the Enigma Machine that was created as a business venture in the 1920's. The Enigma Machine had a series of rotor wheels where each rotor wheel modified the typed letter to some other letter, which was then reflected (another transformation) and then passed back through the rotor wheels. Plug boards were added at the front (and then, on the way back, at the back) of the machine. The symmetry allowed the encryption and decryption to be done if one had an identical Enigma Machine with the identical initial settings at the other end. Of course, security policies had to be in place and followed for the security of the Enigma machine to be not so easily breakable.

The design of the Enigma Machine had a subtle design weakness, later exploited by the Allies, in that no letter could be encrypted to itself. Thus, the concept of a crib (coined by Alan Turing) was developed. That is, if one could guess at a word or phrase that would be in the message (e.g., near the beginning) then one could narrow down the possibilities to put the decryption within the reach of the mechanical tabulating machines, called bombes. For example, suppose that the crib were "**HELLO**" and the encrypted message appears in the first line, line 0, of the following.

```
0. ???L??OL?????????
1. HELLO
2.  HELLO
3.   HELLO
4.    HELLO
5.     HELLO
6.      HELLO
```

The only position where the clear text "**HELLO**" could occur is in line 4. The other lines would be ruled out as colliding with the encrypted message. In addition, the codebreakers used other methods to get cribs and hints, including capturing ships, learning the habits of operators, etc.

Simulators

There are many online simulators for the Enigma machine.



Figure 1: Enigma simulator (top view)

Links to these simulators are provided to the students. One of the most popular realistic Enigma Machine simulators is at <http://www.conferencemgt.com/cgi-bin/seektrack.exe>. This Enigma Machine appears in figure 2. Once the settings are made to the machine, the sender types the letters on the keyboard and the corresponding encryption (or decryption) appears lit up in the upper keyboard. If any keys are pressed out-of-sequence, there is no easy way to go back once a key is mistyped. The plug settings appear in figure 2.

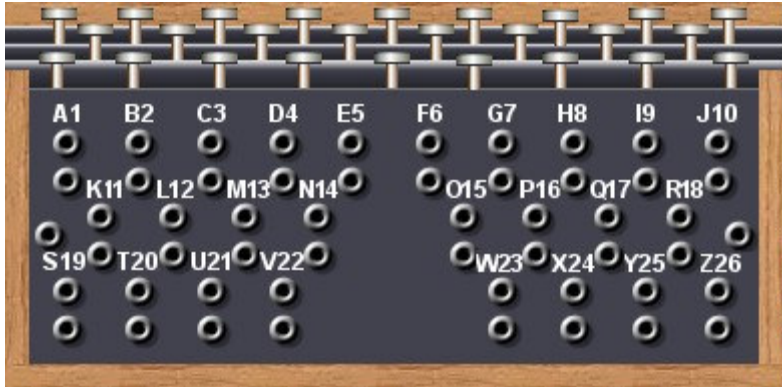


Figure 2: Enigma simulator (plug view)

On the plug board, there are 13 possible cables to connect 26 letters. This provides a very large number of ways to connect the cables (analysis omitted). Under the lid, the rotors can be changed and (internally) rotated. This appears in figure 3.



Figure 3: Enigma simulator (inside view)

In order to generate individualized problems for the students, the author created a functional Enigma simulator. The simulator did not have to be visually accurate as such simulators exist on the internet for download and use. The primary obstacle became one of replicating quirks of the original Enigma machine. This was eventually resolved and only the most common Enigma machine was simulated. The simulator appearing in figure 4 was not provided to the student as it would have made their assignment much easier and not nearly so interesting.

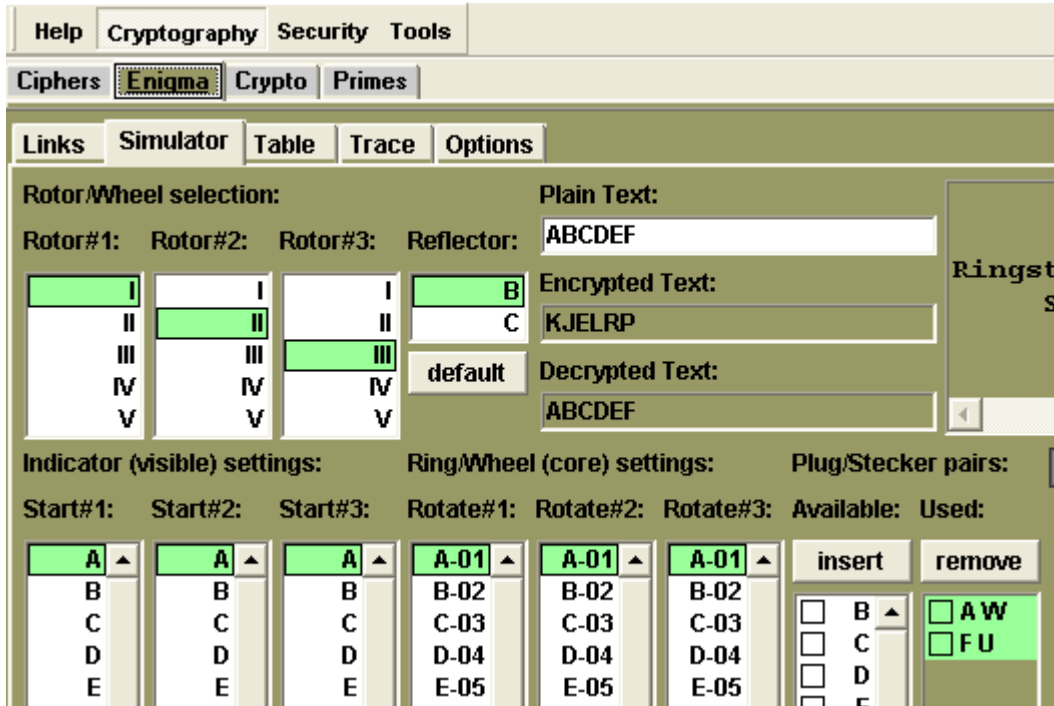


Figure 4: Enigma simulator written by the author

The author used the following method to create individualized problems for the students. First, the initial Enigma Machine settings are generated. For example, the trace of the encryption of the clear text "ABCDEF" to the encrypted text "KJELRP" using the default settings is as follows.

```

Encrypt :
[l:m:r] [l:m:r] .S #3 #2 #1 #R #1 #2 #3 S.
[A:A:A] [l:m:r] AW XS RG GD DH HP PU VL KK
[A:A:B] [l:m:r] BB DH FI IV VW WN NT VL JJ
[A:A:C] [l:m:r] CC FL IX XR RB BW WM PH EE
[A:A:D] [l:m:r] DD HP LH HQ QE EA AA EP LL
[A:A:E] [l:m:r] EE JT OM MO OM MC CP UW RR
[A:A:F] [l:m:r] FU AB VY YC CU UR RG MV PP
[A:A:G]

Decrypt :
[l:m:r] [l:m:r] .S #3 #2 #1 #R #1 #2 #3 S.
[A:A:A] [l:m:r] KK LV UP PH HD DG GR SX WA
[A:A:B] [l:m:r] JJ LV TN NW WV VI IF HD BB
[A:A:C] [l:m:r] EE HP MW WB BR RX XI LF CC
[A:A:D] [l:m:r] LL PE AA AE EQ QH HL PH DD
[A:A:E] [l:m:r] RR WU PC CM MO OM MO TJ EE
[A:A:F] [l:m:r] PP VM GR RU UC CY YV BA UF
[A:A:G]
    
```

2007 ASCUE Proceedings

The above trace is useful when debugging the simulation of the Enigma Machine. On the left, the rotation of each rotor is clearly indicated. The single and double stepping of the rotors when certain notches are encountered must be precisely simulated. Some of the quirks of the machine reduce the total number of possibilities for each encryption.

Here are the default settings, except for the plugs (i.e., Stecker), which are shown as an example.

```
      UKW: B
      Walzen: 1 2 3
Ringstellung: A-01 A-01 A-01
Stecker: AW FU
```

The format used is similar to the settings used in World War II. The above is interpreted as follows, using the most conventional Wehrmacht (i.e., Military) Enigma Machine.

- The reflector is **B** where the choices are **A** or **B**.
- The rotors are, physically from left to right, **I**, **II**, and **III** but the forward pass in the machine is from right to left.
- The rotor settings are all **A-01** where the possibilities are from **A-01** to **Z-26**.
- The plug settings switch characters **A** and **W**, and characters **F** and **U**.

The above settings would be determined and issued secretly well before the day when they were to be used. The initial settings for the rotors that are displayed on top, however, are transmitted at the start of the message. This often became a human issue as some operators become very predictable as to the nonrandom initial settings that they would use.

A message is generated for each student. To keep it simple, the following is the format of the messages.

```
ONEXXTWOXXTHREE
SEVENEIGHTTWOXX
FIVEXTHREENINEX
```

The pattern should be obvious and lets the student know if the decryption is correct (with very high probability). Note that the original Enigma Machine supported only the **26** letters of the alphabet. The letter **'x'** was used for a space. Other abbreviations (omitted) were used to reduce the size of the encoded/decoded messages.

To make the exercise general, **1,000** problem sets from **000** to **999** are generated. An XML file is created for this purpose. The relevant fields are the settings text (i.e., the part that forms the question) and the expected answer text (i.e., that must match exactly).

Each student is given a series of, say, **5** messages to decrypt. These are selected, at random, from the **1,000** problem sets. At the end of the **5** messages, the student must write a paragraph to answer a question. The answer is then submitted to the online database. Later, the submission is graded by the teacher. The results are then made available to the student via the online database (after login, of course).

Here is an example of the XML where 1,000 problem sets are generated. Some details are omitted for space reasons.

```
<?xml version="1.0" standalone="yes"?>
<rmsTable ext="" seed="741" content="">
<row index="1"
  input="PBG BTLGRBMMKUGPRUHZZKW"
  output="TWOXXXTENXXXTHREEX"
  data="Machine settings:
  &lt;br&gt;UKW: B
  &lt;br&gt;Walzen: 4 1 5
  &lt;br&gt;Ringstellung: T-20 Z-26 S-19
  &lt;br&gt;Stecker: BX DH KM"
  mark="0"
/>
<!-- and so on ... -->
</rmsTable>
```

Classes of students are then mapped to the problem sets. Here is an example of the XML. Again, some details are omitted for space reasons.

```
<?xml version="1.0" standalone="yes"?>
<rmsWorks steps="4" parts="1" file="0" content="" >
<account type="teacher" sid="99999" login="..." name="...">
<work index="0" general="Encrypted message to decode">
<part index="0"
  question="LVU BEKQNGSFEWLDQGWUJT"
  answer="SIXXXXTENXXXNINEXX"
  extra="Machine settings:
  &lt;br&gt;UKW: B
  &lt;br&gt;Walzen: 3 5 4
  &lt;br&gt;Ringstellung: N-14 P-16 D-04
  &lt;br&gt;Stecker: BL CP HM"
  file=""
/>
</work>
<!-- and so on ... />
</account>
<!-- and so on ... -->
</rmsWorks>
```

Note that each step could consist of multiple parts (e.g., expected answers), but this example has only one part per step. Since each student has a different problem, copying is not easy. Instead, the only way for a student to help another student is to actually do the work for that student (which may or may not result in the student learning the work from the student helping).

The web-based ASP code permits the use of the exercises, encoded in an XML format, with a document customized to the particular exercise (e.g., with links to help, explanations, etc.). The first step in the Enigma decryption requirement appears in figure 5.

Asmt#5: Enigma decryption (due/on 2006/10/20 , 20 points)	
When ready, select <input type="button" value="Submit"/> .	
Goal	<input type="text" value="Encrypted message to decode"/>
Encrypted text	<input type="text" value="LVU BEKQNGSFEWLDQGVUJT"/>
Decrypted text	<input type="text"/>
You are teacher Snyder, Robin [snyderr]. You are on step 1 of 5 .	
<input type="button" value="Copy demo for teacher"/>	
(expected value)	<input type="text" value="SIXXXXTENXXXNINEXX"/>
Extra data:	Machine settings:
	UKW: B
	Walzen: 3 5 4
	Ringstellung: N-14 P-16 D-04
	Stecker: BL CP HM

Figure 5: First step in the decryption requirement

For demonstration purposes, the teacher has quick access to the actual answer (i.e., the button for "Copy demo for teacher"). The students, though, do not have such access. In addition, to answer the inevitable student questions, the teacher has quick web access to all problem steps and answers for every student.

At exam time, the student must do some of the exercises (this is but one example) in real time. Instead of 5 of the same exercise, however, it may be 5 different exercises. Since the XML records only the question explanation text, expected answer text (and optionally some extra text), the ASP (or PHP) code that runs the web-based interface need not know about the details of the problem. The code can just present the question explanation and let the student proceed if the expected answer text matches.

Videos

The author has been experimenting with audio-video using Camtasia Studio. In the case of this assignment, a demo in class is done while capturing the audio and the relevant parts of the screen. The author has developed a custom interface to act as a decision support system with Camtasia Studio to help manage, produce, and publish the videos for student use.

Summary

This paper has presented an overview of this history in general and the Enigma machine in particular. The author has written an Enigma simulator used to generate problems for students.

References

- [1] Kahn, D. (1996). The Codebreakers : The Comprehensive History of Secret Communication from Ancient Times to the Internet. Scribner.
- [2] Keegan, J. (2003). Intelligence in War: Knowledge of the Enemy from Napoleon to Al-Qaeda. Hutchinson Radius.
- [3] Singh, S. (2000). The Code Book: The Science of Secrecy from Ancient Egypt to Quantum Cryptography. New York: Anchor.
- [4] Snyder, R. (2007). Creating individual student assignments in the historical context of wireless security and the Enigma machine 1st Computer Security Conference (April 12-13, 2007), Myrtle Beach, SC.
- [5] Snyder, R. (2007). Simulating the Enigma machine: Creating customized student assignments. 37th Annual Meeting of the Southeastern Region of the Decision Sciences Institute (February 21-23, 2007), Savannah, GA.
- [6] Snyder, R. (2006). Ethical hacking and password cracking: A pattern for individualized security exercises. 2006 Information Security Curriculum Development conference (September 22-23, 2006), Kennesaw, GA.