NEMA - NEue MAschine

Christoph Ehret,
Jacek Jonczy,
Jürg Nietlispach,
and Nicolas Zwahlen

Krypto WK 2007

September 6, 2007

Outline

Outline

Outline

Outline

Outline

Outline

Outline

Some History

- Swiss cipher machine
- Replaces the Swiss K machine (CH Enigma version)
- Based on German ENIGMA
- First functional model: 1944
- Formal military approval: 1945
- First machine in service: 1947
- Used until around 1970ies
- Declassified 1992
- Supposed number of produced units: 640

Some History

- Swiss cipher machine
- Replaces the Swiss K machine (CH Enigma version)
- Based on German ENIGMA
- First functional model: 1944
- Formal military approval: 1945
- First machine in service: 1947
- Used until around 1970ies
- Declassified 1992
- Supposed number of produced units: 640

Some History

- Swiss cipher machine
- Replaces the Swiss K machine (CH Enigma version)
- Based on German ENIGMA
- First functional model: 1944
- Formal military approval: 1945
- First machine in service: 1947
- Used until around 1970ies
- Declassified 1992
- Supposed number of produced units: 640

Some History

- Swiss cipher machine
- Replaces the Swiss K machine (CH Enigma version)
- Based on German ENIGMA
- First functional model: 1944
- Formal military approval: 1945
- First machine in service: 1947
- Used until around 1970ies
- Declassified 1992
- Supposed number of produced units: 640

Some History

- Swiss cipher machine
- Replaces the Swiss K machine (CH Enigma version)
- Based on German ENIGMA
- First functional model: 1944
- Formal military approval: 1945
- First machine in service: 1947
- Used until around 1970ies
- Declassified 1992
- Supposed number of produced units: 640

Some History

- Swiss cipher machine
- Replaces the Swiss K machine (CH Enigma version)
- Based on German ENIGMA
- First functional model: 1944
- Formal military approval: 1945
- First machine in service: 1947
- Used until around 1970ies
- Declassified 1992
- Supposed number of produced units: 640

Some History

- Swiss cipher machine
- Replaces the Swiss K machine (CH Enigma version)
- Based on German ENIGMA
- First functional model: 1944
- Formal military approval: 1945
- First machine in service: 1947
- Used until around 1970ies
- Declassified 1992
- Supposed number of produced units: 640

Some History

- Swiss cipher machine
- Replaces the Swiss K machine (CH Enigma version)
- Based on German ENIGMA
- First functional model: 1944
- Formal military approval: 1945
- First machine in service: 1947
- Used until around 1970ies
- Declassified 1992
- Supposed number of produced units: 640

Some History

- Swiss cipher machine
- Replaces the Swiss K machine (CH Enigma version)
- Based on German ENIGMA
- First functional model: 1944
- Formal military approval: 1945
- First machine in service: 1947
- Used until around 1970ies
- Declassified 1992
- Supposed number of produced units: 640

Want to buy one?

- Enigma:
  - 6200 USD: private sale [NO] to UK collector: 4000 [A2621]: September 2002
    ⋮
  - 51100 USD: auction [eBay], M4 [M18360], September 2000
- Nema:
  - 5250 USD: auction [eBay], May 2000
    ⋮
  - 845 USD: Auction [Christies, London], July 2000

source: `http://www.eclipse.net/~dhamer/enigma_p.htm`

Outline

NeMa components

- Rotor (wheel) based, like Enigma
- Elements:
  - "Scrambler" unit containing:
    - 10 wheels (5 contact and 5 drivewheels)
    - 6 notch rings
    - stepping levers
  - Keyboard (input)
  - Lampboard (output)

NeMa components

- Rotor (wheel) based, like Enigma
- Elements:
  - "Scrambler" unit containing:
    - 10 wheels (5 contact and 5 drivewheels)
    - 6 notch rings
    - stepping levers
  - Keyboard (input)
  - Lampboard (output)

Encryption components

- **The wheels** - each containing an alphabet- & toothring:
  - There are two types of wheels:
    - **Drive wheels including (ETW)** which perform mechanical encryption
    - **Contact wheels including (UKW)** which perform electronical encryption
  - **Umkehrwalze (UKW)** reflects the signal and performs e. encryption
    $\longrightarrow$ simplifies the decryption
  - **Eintrittswalze (ETW)** performs additionally mechanical encryption

Encryption components

- **The wheels** - each containing an alphabet- & toothring:
  - There are two types of wheels:
    - **Drive wheels including (ETW)** which perform mechanical encryption
    - **Contact wheels including (UKW)** which perform electronical encryption
  - **Umkehrwalze (UKW)** reflects the signal and performs e. encryption
    $\longrightarrow$ simplifies the decryption
  - **Eintrittswalze (ETW)** performs additionally mechanical encryption

Encryption components

- **The wheels** - each containing an alphabet- & toothring:
  - There are two types of wheels:
    - **Drive wheels including (ETW)** which perform mechanical encryption
    - **Contact wheels including (UKW)** which perform electronical encryption
  - **Umkehrwalze (UKW)** reflects the signal and performs e. encryption
    $\longrightarrow$ simplifies the decryption
  - **Eintrittswalze (ETW)** performs additionally mechanical encryption

Encryption components

- **The wheels** - each containing an alphabet- & toothring:
  - There are two types of wheels:
    - **Drive wheels including (ETW)** which perform mechanical encryption
    - **Contact wheels including (UKW)** which perform electronical encryption
  - **Umkehrwalze (UKW)** reflects the signal and performs e. encryption
    $\longrightarrow$ simplifies the decryption
  - **Eintrittswalze (ETW)** performs additionally mechanical encryption

Encryption components

- **The wheels** - each containing an alphabet- & toothring:
  - There are two types of wheels:
    - **Drive wheels including (ETW)** which perform mechanical encryption
    - **Contact wheels including (UKW)** which perform electronical encryption
  - **Umkehrwalze (UKW)** reflects the signal and performs e. encryption
    $\longrightarrow$ simplifies the decryption
  - **Eintrittswalze (ETW)** performs additionally mechanical encryption

Encryption components

- **The wheels** - each containing an alphabet- & toothring:
  - There are two types of wheels:
    - **Drive wheels including (ETW)** which perform mechanical encryption
    - **Contact wheels including (UKW)** which perform electronical encryption
  - **Umkehrwalze (UKW)** reflects the signal and performs e. encryption
    $\longrightarrow$ simplifies the decryption
  - **Eintrittswalze (ETW)** performs additionally mechanical encryption

Encryption components, cont.

- **Stepping levers:** Cause the rotation of the wheels
- **Entry plate (hidden):** The current enters the scrambler on (different) channels whenever a button is stroked. Each button has its own entry channel: port1 - Q (top), port2 - W, port3 - E, ..., port 26 - M

Encryption components, cont.

- **Stepping levers:** Cause the rotation of the wheels
- **Entry plate (hidden):** The current enters the scrambler on (different) channels whenever a button is stroked. Each button has its own entry channel: port1 - Q (top),
port2 - W, port3 - E, ..., port 26 - M

Electrical encryption with contact wheels

- Permutations are realized with "random" wiring of inports and outports

	Y	Z	A	B	C	D	E	F	G	H
Wheel	17	18	19	20	21	22	23	24	25	26
A	08	20	06	11	03	01	12	21	07	17
B	14	21	03	02	17	06	12	05	23	26
⋮										
F					⋱					
UKW										
ETW	02	03	04	17	16	21	14	20	06	05

Drive wheels

- Each drive wheel has a notch ring (pattern)
- Notch ring has high regions (1) and low regions (0)

N.R.	S	T	U	V	W	X	Y	Z
1	1	0	0	0	0	0	1	1
2	0	0	0	0	0	0	0	0
12	1	0	1	1	1	1	1	1
⋮								
22				⋱				
23								

Drive wheels

- Each drive wheel has a notch ring (pattern)
- Notch ring has high regions (1) and low regions (0)

N.R.	S	T	U	V	W	X	Y	Z
1	1	0	0	0	0	0	1	1
2	0	0	0	0	0	0	0	0
12	1	0	1	1	1	1	1	1
⋮								
22				⋱				
23								

Stepping levers

- Stepping levers cause the rotation of the wheels. There are two types of stepping levers:
  - The **toothring lever** grabs the toothring
  - The **notchring lever** touchs the notchring and grab the (neighbouring) toothring
- A pair of toothring and notchring levers is combined to an "arm"
- "Arms" can (initially) be blocked (part of the key)

Stepping levers

- Stepping levers cause the rotation of the wheels. There are two types of stepping levers:
  - The **toothring lever** grabs the toothring
  - The **notchring lever** touchs the notchring and grab the (neighbouring) toothring
- A pair of toothring and notchring levers is combined to an "arm"
- "Arms" can (initially) be blocked (part of the key)

Stepping levers

- Stepping levers cause the rotation of the wheels. There are two types of stepping levers:
  - The **toothring lever** grabs the toothring
  - The **notchring lever** touchs the notchring and grab the (neighbouring) toothring
- A pair of toothring and notchring levers is combined to an "arm"
- "Arms" can (initially) be blocked (part of the key)

Mechanical encryption

The mechanical encryption works as follows:

- Every wheel always rotate one position per each keystroke
  $\longrightarrow$ polyalphabetic substitution
- Additionally encryption with (arbitrary) rotation of contactwheels
  $\longrightarrow$ rotation depending on antecedent drive wheel (notchring pattern)
  If notchring region was 1$\longrightarrow$ stepping levers arised $\longrightarrow$ toothringlever can not grab $\longrightarrow$ wheel desn't move (and vice versa if notchring region was 0)
- More encryption with the right notch ring of the ETW
  $\longrightarrow$ whole arms can be controlled
  The arms are arised if the notchring region was 1 $\longrightarrow$ both wheels don't move

Mechanical encryption

The mechanical encryption works as follows:

- Every wheel always rotate one position per each keystroke
  $\longrightarrow$ polyalphabetic substitution
- Additionally encryption with (arbitrary) rotation of contactwheels
  $\longrightarrow$ rotation depending on antecedent drive wheel (notchring pattern)
  If notchring region was 1$\longrightarrow$ stepping levers arised $\longrightarrow$ toothringlever can not grab $\longrightarrow$ wheel desn't move (and vice versa if notchring region was 0)
- More encryption with the right notch ring of the ETW
  $\longrightarrow$ whole arms can be controlled
  The arms are arised if the notchring region was 1 $\longrightarrow$ both wheels don't move

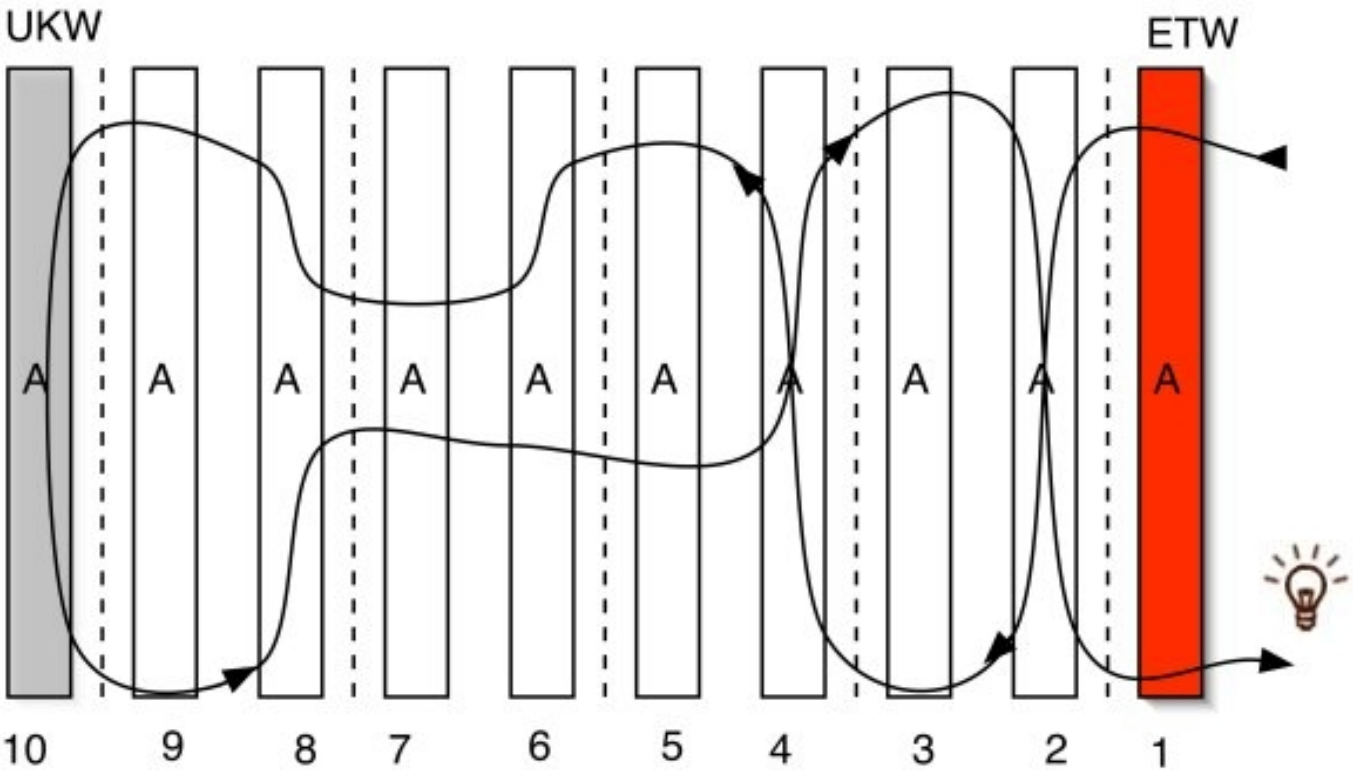
Mechanical encryption

The mechanical encryption works as follows:

- Every wheel always rotate one position per each keystroke
  $\longrightarrow$ polyalphabetic substitution
- Additionally encryption with (arbitrary) rotation of contactwheels
  $\longrightarrow$ rotation depending on antecedent drive wheel (notchring pattern)
  If notchring region was 1$\longrightarrow$ stepping levers arised $\longrightarrow$ toothringlever can not grab $\longrightarrow$ wheel desn't move (and vice versa if notchring region was 0)
- More encryption with the right notch ring of the ETW
  $\longrightarrow$ whole arms can be controlled
  The arms are arised if the notchring region was 1 $\longrightarrow$ both wheels don't move

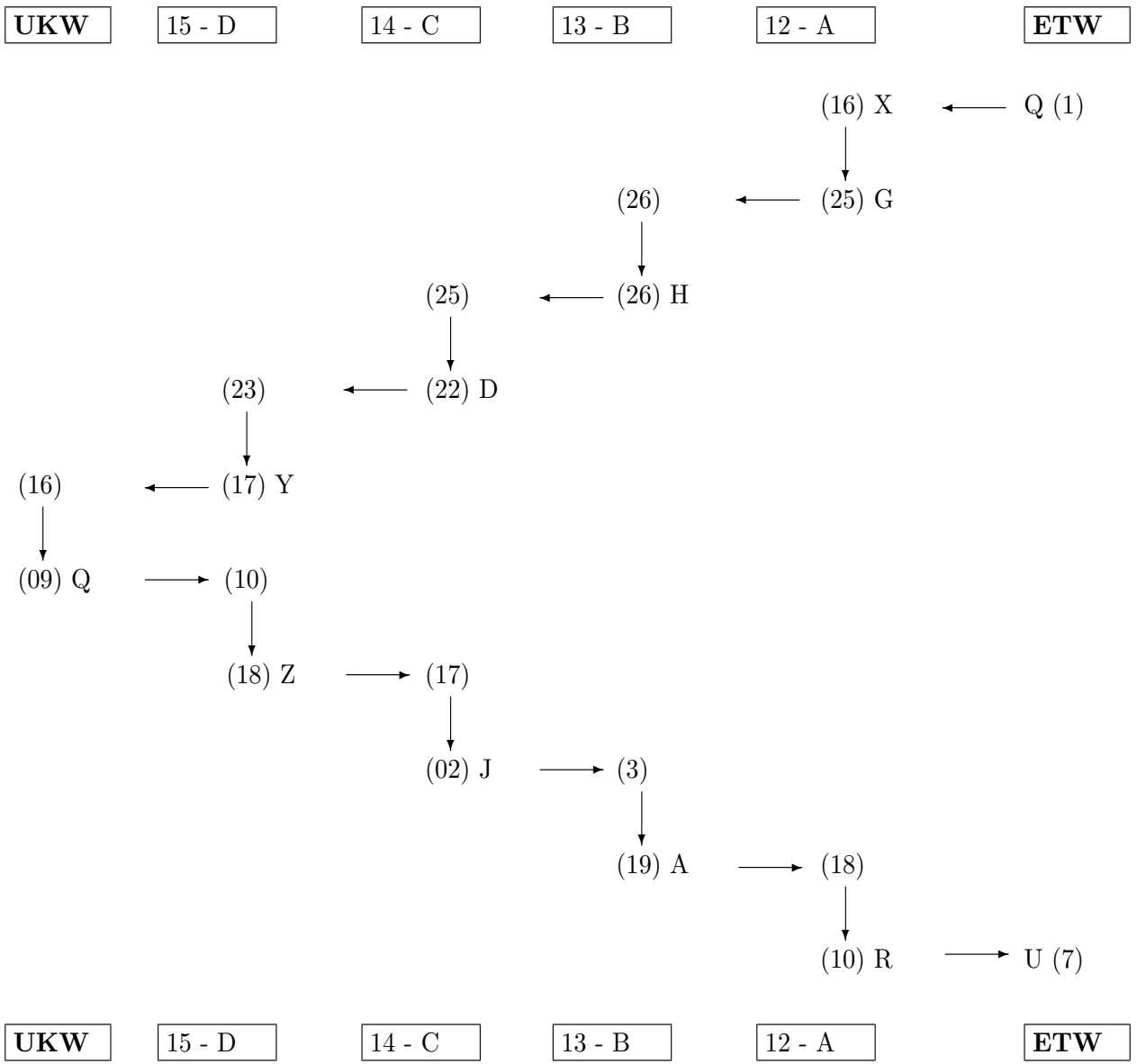
Track of the electrical signal

Current flow through wheels - entering through the entry plate - for the encryption:



In case of decryption the current flows in the reverse direction (with the same configuration)

Track of the electrical signal, cont.



Main differences between NeMa and ENIGMA

- ENIGMA has a "clockwork" motion of the wheels, the NeMa motion is (mostly) random
- ENIGMA substitutes/redirects some channels between the keyboard and the entry plate (via plugboard), NeMa has a "static" wiring

Outline

Cryptographic Properties

- Polyalphabetic stream cipher
- Two keys:
  - **Inner key** (Wochenschlüssel): initial machine configuration:

$$r_1\, C_1 \mid r_2\, C_2 \mid r_3\, C_3 \mid r_4\, C_4$$

  - **Outer key** (Tagesschlüssel): 10-character-string, generated from secret *code word* and *random letters*
- Inner key space: $360 \cdot 212520 \approx 2^{26}$
- Outer key space: $26^{10} \approx 2^{47}$
- Encryption and decryption:

$$E_{k_{in},k_{out}}(m) = P_1 \circ P_2 \circ P_3 \circ P_4 \circ P_5 \circ P_4 \circ P_3 \circ P_2 \circ P_1(m) = c$$
$$D_{k_{in},k_{out}}(c) = P_1 \circ P_2 \circ P_3 \circ P_4 \circ P_5 \circ P_4 \circ P_3 \circ P_2 \circ P_1(c) = m$$

Cryptographic Properties

- Polyalphabetic stream cipher
- Two keys:
  - **Inner key** (Wochenschlüssel): initial machine configuration:

$$r_1 C_1 \mid r_2 C_2 \mid r_3 C_3 \mid r_4 C_4$$

  - **Outer key** (Tagesschlüssel): 10-character-string, generated from secret *code word* and *random letters*
- Inner key space: $360 \cdot 212520 \approx 2^{26}$
- Outer key space: $26^{10} \approx 2^{47}$
- Encryption and decryption:

$$E_{k_{in},k_{out}}(m) = P_1 \circ P_2 \circ P_3 \circ P_4 \circ P_5 \circ P_4 \circ P_3 \circ P_2 \circ P_1(m) = c$$
$$D_{k_{in},k_{out}}(c) = P_1 \circ P_2 \circ P_3 \circ P_4 \circ P_5 \circ P_4 \circ P_3 \circ P_2 \circ P_1(c) = m$$

Cryptographic Properties

- Polyalphabetic stream cipher
- Two keys:
  - **Inner key** (Wochenschlüssel): initial machine configuration:

$$r_1 C_1 \mid r_2 C_2 \mid r_3 C_3 \mid r_4 C_4$$

  - **Outer key** (Tagesschlüssel): 10-character-string, generated from secret *code word* and *random letters*
- Inner key space: $360 \cdot 212520 \approx 2^{26}$
- Outer key space: $26^{10} \approx 2^{47}$
- Encryption and decryption:

$$E_{k_{in},k_{out}}(m) = P_1 \circ P_2 \circ P_3 \circ P_4 \circ P_5 \circ P_4 \circ P_3 \circ P_2 \circ P_1(m) = c$$
$$D_{k_{in},k_{out}}(c) = P_1 \circ P_2 \circ P_3 \circ P_4 \circ P_5 \circ P_4 \circ P_3 \circ P_2 \circ P_1(c) = m$$

Cryptographic Properties

- Polyalphabetic stream cipher
- Two keys:
  - **Inner key** (Wochenschlüssel): initial machine configuration:

$$r_1 C_1 \mid r_2 C_2 \mid r_3 C_3 \mid r_4 C_4$$

  - **Outer key** (Tagesschlüssel): 10-character-string, generated from secret *code word* and *random letters*
- Inner key space: $360 \cdot 212520 \approx 2^{26}$
- Outer key space: $26^{10} \approx 2^{47}$
- Encryption and decryption:

$$E_{k_{in},k_{out}}(m) = P_1 \circ P_2 \circ P_3 \circ P_4 \circ P_5 \circ P_4 \circ P_3 \circ P_2 \circ P_1(m) = c$$
$$D_{k_{in},k_{out}}(c) = P_1 \circ P_2 \circ P_3 \circ P_4 \circ P_5 \circ P_4 \circ P_3 \circ P_2 \circ P_1(c) = m$$

Cryptographic Properties

- Polyalphabetic stream cipher
- Two keys:
  - **Inner key** (Wochenschlüssel): initial machine configuration:

$$r_1 C_1 \mid r_2 C_2 \mid r_3 C_3 \mid r_4 C_4$$

  - **Outer key** (Tagesschlüssel): 10-character-string, generated from secret *code word* and *random letters*
- Inner key space: $360 \cdot 212520 \approx 2^{26}$
- Outer key space: $26^{10} \approx 2^{47}$
- Encryption and decryption:

$$E_{k_{in},k_{out}}(m) = P_1 \circ P_2 \circ P_3 \circ P_4 \circ P_5 \circ P_4 \circ P_3 \circ P_2 \circ P_1(m) = c$$
$$D_{k_{in},k_{out}}(c) = P_1 \circ P_2 \circ P_3 \circ P_4 \circ P_5 \circ P_4 \circ P_3 \circ P_2 \circ P_1(c) = m$$

Generating the outer key (message key)

1. Choose code word, set on machine
2. Choose 10 random letters, place at beginning and end of ciphertext
3. Encrypt random letters using initial setting
4. Result: secret message key used for encryption

Generating the outer key (message key)

1. Choose code word, set on machine
2. Choose 10 random letters, place at beginning and end of ciphertext
3. Encrypt random letters using initial setting
4. Result: secret message key used for encryption

Generating the outer key (message key)

1. Choose code word, set on machine
2. Choose 10 random letters, place at beginning and end of ciphertext
3. Encrypt random letters using initial setting
4. Result: secret message key used for encryption

Generating the outer key (message key)

1. Choose code word, set on machine
2. Choose 10 random letters, place at beginning and end of ciphertext
3. Encrypt random letters using initial setting
4. Result: secret message key used for encryption

Strengths

- Huge key space ($\sim 2^{73}$)
- Complex (tricky) stepping
- Rather large cycle length: 17576 (max)
  $\longrightarrow$ index of coincidence method useless
- Linguistic cryptanalysis useless

Weaknesses

- Exploitable incautious usage:
  - inner keys rarely changed
  - codewords often dictionary terms
  - redundancies in cleartext
  - probable words (cf. Enigma, *cribs*)
  - etc.
- Exploitable weaknesses:
  - reflector wheel (cf. Enigma) $\longrightarrow$ involutions
  - not whole inner key space must be searched
  - other (?)

Weaknesses

- Exploitable incautious usage:
  - inner keys rarely changed
  - codewords often dictionary terms
  - redundancies in cleartext
  - probable words (cf. Enigma, *cribs*)
  - etc.
- Exploitable weaknesses:
  - reflector wheel (cf. Enigma) $\longrightarrow$ involutions
  - not whole inner key space must be searched
  - other (?)

Outline

Brute-Force Attack

- Try all 26^{10} keys using a C++ simulation
- Identify english plaintext based on a χ^2 test
  - finds correct key
  - 1 wrong match per 4'000'000 keys
- Rate: 31'300 keys/second (52 character ciphertext, 1.86 GHz processor)
- Time needed for all keys: about 145 years
- (At this rate, a brute-force attack on a 4-wheel Enigma would take about 6 minutes)

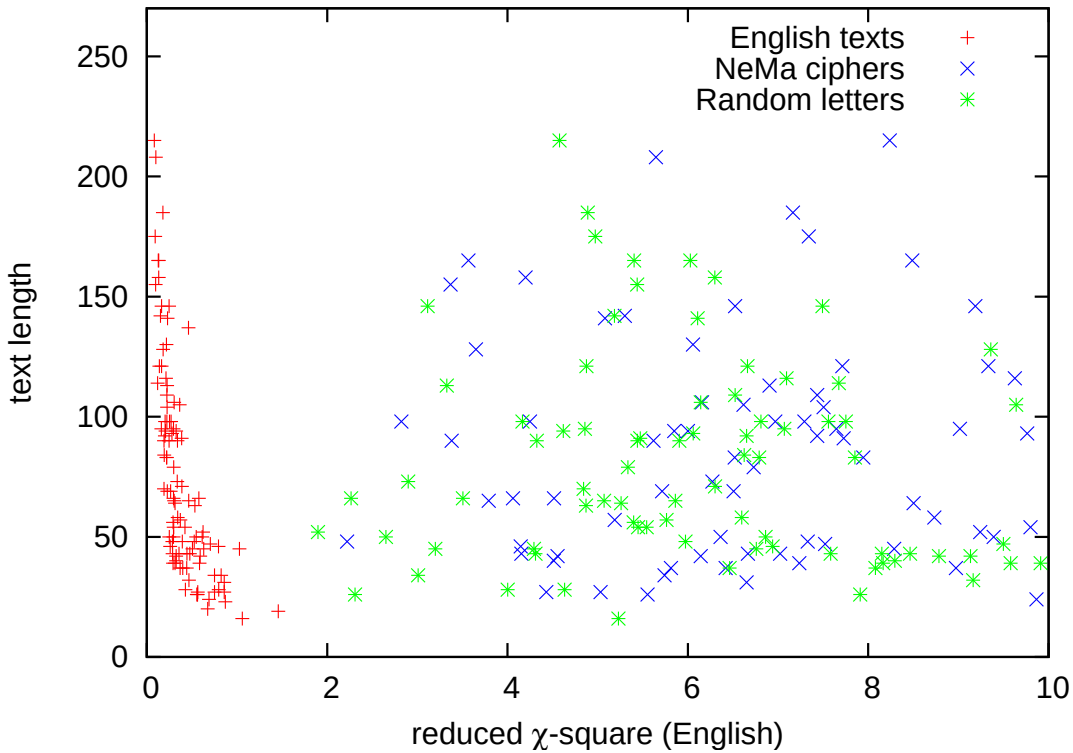
Brute-Force Attack: identify English plaintext

Compute a χ^2 based on the letter frequencies f_i of the English language:

$$\chi^2 = \sum_{i=0}^{25} \left(\frac{N_i - Nf_i}{\sqrt{Nf_i}} \right)^2$$

- $\chi^2/N < 0.3$ for English texts
- $\chi^2/N > 0.4$ for random letters

Need at least $N = 50$ characters, not more



Known-Plaintext Attack

- Find key given plaintext and ciphertext
- Also try all 26^{10} keys
- Decrypt ciphertext letter by letter
  - reject key as soon as there is a contradiction with the plaintext
  - no wrong matches
- Rate: 480'000 keys/second (independent on ciphertext length, 1.86 GHz processor)
- Time needed for all keys: about 13 years

Outline

NEMA Simulations

- A simulation running on Windows already existed (by F. Weierud, with GUI)
- We developped a multi-platform simulation in `perl`, as a prototype,
- and a `C++` simulation for attacks (50 times faster than perl...)

Outline

Conclusion & Outlook

- NEMA successfully replaced the Swiss Enigma machine
- It was cryptographically well designed and constructed
- The complexity of NEMA is superior to that of Enigma
- The best attack we could find is brute-force and it resists very well to it
- Swiss cryptographers have done a good job ... as always ;-)
- Outlook :
  - Improve our simulation with a graphical user interface
  - Try to find out some more sophisticated attacks

Conclusion & Outlook

- NEMA successfully replaced the Swiss Enigma machine
- It was cryptographically well designed and constructed
- The complexity of NEMA is superior to that of Enigma
- The best attack we could find is brute-force and it resists very well to it
- Swiss cryptographers have done a good job ... as always ;-)
- Outlook :
  - Improve our simulation with a graphical user interface
  - Try to find out some more sophisticated attacks

Conclusion & Outlook

- NEMA successfully replaced the Swiss Enigma machine
- It was cryptographically well designed and constructed
- The complexity of NEMA is superior to that of Enigma
- The best attack we could find is brute-force and it resists very well to it
- Swiss cryptographers have done a good job ... as always ;-)
- Outlook :
  - Improve our simulation with a graphical user interface
  - Try to find out some more sophisticated attacks

Conclusion & Outlook

- NEMA successfully replaced the Swiss Enigma machine
- It was cryptographically well designed and constructed
- The complexity of NEMA is superior to that of Enigma
- The best attack we could find is brute-force and it resists very well to it
- Swiss cryptographers have done a good job ... as always ;-)
- Outlook :
  - Improve our simulation with a graphical user interface
  - Try to find out some more sophisticated attacks

Conclusion & Outlook

- NEMA successfully replaced the Swiss Enigma machine
- It was cryptographically well designed and constructed
- The complexity of NEMA is superior to that of Enigma
- The best attack we could find is brute-force and it resists very well to it
- Swiss cryptographers have done a good job ... as always ;-)
- Outlook :
  - Improve our simulation with a graphical user interface
  - Try to find out some more sophisticated attacks

Conclusion & Outlook

- NEMA successfully replaced the Swiss Enigma machine
- It was cryptographically well designed and constructed
- The complexity of NEMA is superior to that of Enigma
- The best attack we could find is brute-force and it resists very well to it
- Swiss cryptographers have done a good job ... as always ;-)
- Outlook :
  - Improve our simulation with a graphical user interface
  - Try to find out some more sophisticated attacks

Conclusion & Outlook

- NEMA successfully replaced the Swiss Enigma machine
- It was cryptographically well designed and constructed
- The complexity of NEMA is superior to that of Enigma
- The best attack we could find is brute-force and it resists very well to it
- Swiss cryptographers have done a good job ... as always ;-)
- Outlook :
  - Improve our simulation with a graphical user interface
  - Try to find out some more sophisticated attacks

Thanks for your attention