3.14 Nicht-periodische Schlüssel mittels Auto-Key

- komplett nicht-periodische Schlüssel lassen sich erreichen, wenn der Text in die Schlüsselerzeugung einbezogen wird
- grundsätzlich problematisch: Verschleppung von Chiffrierfehlern
- einfache Auto-Key-Verfahren oft leicht angreifbar
- Beispiel:
  - VIGENÈRE-Verschlüsselung
  - erste d Klartextzeichen $x_1 \dots x_d$ werden mit geheimem Schlüsselkeim $c_1 \dots c_d$ chiffriert
  - danach wird i-tes Zeichen mit $c_i = x_{i-d}$ chiffriert
  - es gilt jedoch

$$\begin{aligned} y_i &= x_i + c_i && \text{für } 1 \leq i \leq d \\ y_{i+d} - y_i &= x_{i+d} - c_i \\ y_{i+2d} - y_{i+d} + y_i &= x_{i+2d} + c_i \\ y_{i+3d} - y_{i+2d} + y_{i+d} - y_i &= x_{i+3d} - c_i \\ &\vdots \end{aligned}$$

  - effektiv also lediglich Verdoppelung der Periode, mit abhängigen Schlüsseln der beiden Hälften
- ähnliche Ansätze als Stromchiffren aber durchaus effektiv

4 Polygraphische Substitution

- polygraphisch: mehrere Klartextzeichen werden auf einmal verschlüsselt
- Anzahl möglicher Substitutionen steigt rasant; bei Verschlüsselung von n Zeichen aus einem Alphabet der Mächtigkeit $N = 26$:

n	1	2	3
$(N^n)!$	$4{,}03 \cdot 10^{26}$	$1{,}88 \cdot 10^{1621}$	$1{,}19 \cdot 10^{66\,978}$

- allgemeine Substitution (keine Systematik) nur bis etwa $n = 3$ handhabbar (Tabelle mit $26^3 = 17576$ Einträgen)

4.1 Bigramm-Substitution

- es werden Bigramme substituiert, also $\chi : V^2 \longrightarrow W^{(m)}$
- Spezielfall Bigramm-Permutation, also $\chi : V^2 \longrightarrow V^2$
- Angabe durch Tabelle

	a	b	c	d	e	f	g	h	i	j	...
a	CA	FN	BL	OU	IH	OO	IL	BV	BE	ER	...
b	SK	WM	DG	IA	CW	PF	IF	VD	DA	XZ	...
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋱

- falls nicht involutorisch: inverse Tabelle nötig
- ab 1. Mai 1937 von deutscher Marine verwendet, um Enigma-Spruchschlüssel überzuchiffrieren; aber nur zehn (involutorische) Tabellen in Gebrauch, die von den Briten aufgeklärt wurden
- Angriff gegen Bigramm-Substitutionen über Bigramm-Häufigkeiten, entsprechend Zeichenhäufigkeiten bei monographischen Verfahren

- PLAYFAIR-Verfahren: Bigramm-Permutation über Alphabet mit $N = 25$ Zeichen (natürliches Alphabet ohne j)
- wurde noch im zweiten Weltkrieg (teilweise leicht modifiziert) auf unterer Kommandoebene benutzt
- basiert auf einem 5×5-Quadrat, beispielsweise

S	C	H	L	U
E	T	X	A	B
D	F	G	I	K
M	N	O	P	Q
R	V	W	Y	Z

- falls beide Bigramm-Zeichen in einer Zeile: beide auf rechten Nachbarn abbilden (mit Ringschluss); hu $\mapsto$ LS
- falls beide Bigramm-Zeichen in einer Spalte: beide auf unteren Nachbarn abbilden (mit Ringschluss); qu $\mapsto$ ZB
- sonst Überkreuzschritt: neuer Buchstabe in selber Zeile, aber Spalte des jeweils anderen; da $\mapsto$ IE
- doppelte Buchstaben werden durch Einfügen von x aufgelöst; tasse $\mapsto$ tasxse $\mapsto$ XBHEED

- kombinatorische Komplexität von PLAYFAIR relativ gering
  - nur $25! = 1{,}55 \cdot 10^{25}$ mögliche 5×5-Quadrate
  - davon jeweils 25 durch zyklische Verschiebung gebildete äquivalent

S	C	H	L	U		W	Y	Z	R	V
E	T	X	A	B		H	L	U	S	C
D	F	G	I	K	$\Leftrightarrow$	X	A	B	E	T
M	N	O	P	Q		G	I	K	D	F
R	V	W	Y	Z		O	P	Q	M	N

  - also nur $24! = 6{,}20 \cdot 10^{23}$ Möglichkeiten – weniger als monographische Substitution
- Angriff über Bigramm-Häufigkeiten oder wahrscheinliches Wort, zusammen mit der Einschränkung der möglichen Substitutionen $\Rightarrow$ falsche Annahmen führen schnell zu Widerspruch

4.2 Codes

- Codes besitzen einen Chiffrierschritt $\chi : C \longrightarrow W^m, C \in V^*$ variabler Breite für eine Untermenge von V^*
- C so gewählt, dass sich jeder Klartext in Elemente von C zerlegen lässt (in der Regel $V \subset C$)
- häufige Silben, Wörter Phrasen in C enthalten
- Wahl möglichst langer Segmente bei der Codierung
- einteilige Codes erhalten die lexikographische Ordnung, um decodierung zu ermöglichen; Beispiel (Auszug aus Signalbuch der deutschen Kaiserlichen Marine, 1914):

63940	OAT	Ohnmacht, -ig
63941	OAU	Ohr, Ohren-
63942	OAÜ	Okkupation, Okkupations-, -ieren
63943	OAV	Ökonomie, -isch
63944	OAW	Oktant

- zweiteilige Codes enthalten für Codierung und Decodierung jeweils passend sortierte Teile

- Codes umfangreich und schwierig zu wechseln – es fehlt der Schlüssel
- zur Geheimhaltung nur in Kombination mit Verschlüsselung zu gebrauchen
- Codierung vor Verschlüsselung (überchiffrierter Code) sinnvoll, falls Code gut gewählt (und diszipliniert eingesetzt)
  - Zeichenhäufigkeiten werden nivelliert
  - wahrscheinliches Wort wird verkürzt, weniger "Angriffsfläche"
  - Nachricht wird verkürzt, weniger "Angriffsfläche" und geringerer Übertragungsaufwand

4.3 Lineare Substitution

- Buchstaben werden wieder isomorph zu Elementen aus $\mathbb{Z}_{26}$ behandelt
- Chiffrierschritt als lineare Abbildung

$$\chi(x) = xT + t, \quad x \in \mathbb{Z}_{26}^n, t \in \mathbb{Z}_{26}^n, T \in \mathbb{Z}_{26}^{n \times n}$$

- falls $T = I$ Reduktion zu VIGENÉRE mit periodischem Schlüssel, $d = n$
- invertierbar, falls T regulär (über $\mathbb{Z}_{26}$)

$$\chi^{-1}(y) = (y - t)T^{-1} = yT^{-1} - tT^{-1}$$

- Konstruktion reguläre Matrix T als $T = LDU$ mit L, U untere bzw. obere Dreiecksmatrix mit Einsen in der Diagonale und D Diagonalmatrix mit invertierbaren Elementen
- involutorisch, falls $(xT + t)T + t = xT^2 + tT + t = x$, also $T^2 = I$ und $tT = -t$

Beispiel:

$$T = \begin{pmatrix} 1 & 0 & 0 \\ 22 & 1 & 0 \\ 12 & 9 & 1 \end{pmatrix} \begin{pmatrix} 3 & 0 & 0 \\ 0 & 19 & 0 \\ 0 & 0 & 17 \end{pmatrix} \begin{pmatrix} 1 & 3 & 7 \\ 0 & 1 & 15 \\ 0 & 0 & 1 \end{pmatrix} \simeq \begin{pmatrix} 3 & 9 & 21 \\ 14 & 9 & 19 \\ 10 & 19 & 0 \end{pmatrix}$$

$$t = \begin{pmatrix} 8 & 0 & 12 \end{pmatrix}$$

$\texttt{krypto} \equiv \begin{pmatrix} 10 & 17 & 24 \end{pmatrix} \begin{pmatrix} 15 & 19 & 14 \end{pmatrix}$

$$\begin{pmatrix} 10 & 17 & 24 \end{pmatrix} \begin{pmatrix} 3 & 9 & 21 \\ 14 & 9 & 19 \\ 10 & 19 & 0 \end{pmatrix} + \begin{pmatrix} 8 & 0 & 12 \end{pmatrix} \simeq \begin{pmatrix} 22 & 23 & 25 \end{pmatrix} \equiv \texttt{WXZ}$$

$$\begin{pmatrix} 15 & 19 & 14 \end{pmatrix} \begin{pmatrix} 3 & 9 & 21 \\ 14 & 9 & 19 \\ 10 & 19 & 0 \end{pmatrix} + \begin{pmatrix} 8 & 0 & 12 \end{pmatrix} \simeq \begin{pmatrix} 17 & 0 & 12 \end{pmatrix} \equiv \texttt{RAM}$$

also Verschlüsselung zu `WXZRAM`

$$T = \begin{pmatrix} 1 & 0 & 0 \\ 22 & 1 & 0 \\ 12 & 9 & 1 \end{pmatrix} \begin{pmatrix} 3 & 0 & 0 \\ 0 & 19 & 0 \\ 0 & 0 & 17 \end{pmatrix} \begin{pmatrix} 1 & 3 & 7 \\ 0 & 1 & 15 \\ 0 & 0 & 1 \end{pmatrix}$$

$$T^{-1} = \begin{pmatrix} 1 & 3 & 7 \\ 0 & 1 & 15 \\ 0 & 0 & 1 \end{pmatrix}^{-1} \begin{pmatrix} 3 & 0 & 0 \\ 0 & 19 & 0 \\ 0 & 0 & 17 \end{pmatrix}^{-1} \begin{pmatrix} 1 & 0 & 0 \\ 22 & 1 & 0 \\ 12 & 9 & 1 \end{pmatrix}^{-1}$$

$$\simeq \begin{pmatrix} 1 & 23 & 12 \\ 0 & 1 & 11 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 9 & 0 & 0 \\ 0 & 11 & 0 \\ 0 & 0 & 23 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 4 & 1 & 0 \\ 4 & 17 & 1 \end{pmatrix} \simeq \begin{pmatrix} 19 & 5 & 16 \\ 16 & 22 & 19 \\ 14 & 1 & 23 \end{pmatrix}$$

$$\left(\begin{pmatrix} 22 & 23 & 25 \end{pmatrix} - \begin{pmatrix} 8 & 0 & 12 \end{pmatrix} \right) \begin{pmatrix} 19 & 5 & 16 \\ 16 & 22 & 19 \\ 14 & 1 & 23 \end{pmatrix} \simeq \begin{pmatrix} 10 & 17 & 24 \end{pmatrix} \equiv \texttt{kry}$$

$$\left(\begin{pmatrix} 17 & 0 & 12 \end{pmatrix} - \begin{pmatrix} 8 & 0 & 12 \end{pmatrix} \right) \begin{pmatrix} 19 & 5 & 16 \\ 16 & 22 & 19 \\ 14 & 1 & 23 \end{pmatrix} \simeq \begin{pmatrix} 15 & 19 & 14 \end{pmatrix} \equiv \texttt{pto}$$

Angriff über wahrscheinliches Wort der Länge $k \cdot n$:

- bei richtiger Lage x und y bekannt in $y = xT + t$ für k verschiedene x und y
- zusammenfassen der Gleichungungen

$$\begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_k \end{pmatrix} = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_k \end{pmatrix} T + \begin{pmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{pmatrix} t = \begin{pmatrix} x_1 & 1 \\ x_2 & 1 \\ \vdots & \vdots \\ x_k & 1 \end{pmatrix} \begin{pmatrix} T \\ t \end{pmatrix}$$

- nach T, t auflösbar, falls $k \geq n + 1$, also Mindestlänge des wahrscheinlichen Wortes $= (n + 1) \cdot n = n^2 + n$
- falls wahrscheinliches Wort länger als nötig: Gleichungssystem überbestimmt; Ausschluss falscher Lage falls keine Lösung

5 Transposition

- die Zeichen des Klartextes werden beibehalten, aber in der Reihenfolge geändert
- einfachste Verfahren schreiben Nachricht z.B. waagerecht in Zeilen vorgegebener Länge lesen Senkrecht heraus

```
m  a  n  k  o  e  n
n  t  e  s  t  a  t
t  d  e  s  s  e  n
a  u  c  h  d  i  a
g  o  n  a  l  h  e
r  a  u  s  l  e  s
e  n
```

  $\Rightarrow$ `MNTAG REATD UOANN EECNU KSSHA SOTSD LLEAE IHENT NAES`
- letzte Zeile wird häufig mit Blendern aufgefüllt – unnötig, da Empfänger Länge der Nachricht kennt