DOI 10.12851/EESJ201406C06ART07

Oleg M. Eliiashiv,
ScM;

Leonid F. Politanskii,
DPh, Professor;

Ruslan L. Politanskii,
Post-Doc, assistant;

Nazariy G. Hladun,
graduate;
Yuriy Fedkovych Chernivtsi National University

Software Implementation of Multi-User Text Messaging System Using Logistic Map

Key words: *Logistic map, software, synchronization, encryption*

Annotation: *The article contains the result of proposed system of information transfer, which can serve up to 20 users on the basis of logistic map generators. Implemented the synchronization process by using checksum verification of system parameters. The structure of the implemented array of data allows it conversion and integration of third-party software.*

The providing of confidential information transfer in multi-user systems is a primary trend in the development of information transfer in computer systems. Implementation of the proposed multi-system data uses block encryption algorithm based on XOR. Formation of encryption keys is based on. The analytical representation of generator is presented in the following form:

$$x_{n+1} = rx_n(1 - x_n) \quad (1)$$

x_{n+1} – the next value of the dynamic variable;

n – iteration step;

r – control parameter;

x_n – the current value of the dynamic variable.

Depending on the parameter value r generated data can be periodic, quasi-periodic or chaotic. From the dependencies of the Lyapunov exponent λ on control parameter (describes the degree of dependence of the dynamic system on the initial conditions and determines the rate of divergence of trajectories in phase space , ranges of parameter values for which there are periodic, quasi- periodic and chaotic oscillations) and the bifurcation

diagram of the logistic map (Figure 1 and Figure 2) implies that using parameter $r \geq 3.56$ Lyapunov exponent λ takes positive values and the period-doubling bifurcation has a high frequency, which indicates the chaotic nature of the system. The initial values of the dynamic variables are selected in the range 0.3 ÷ 0.8 (3).

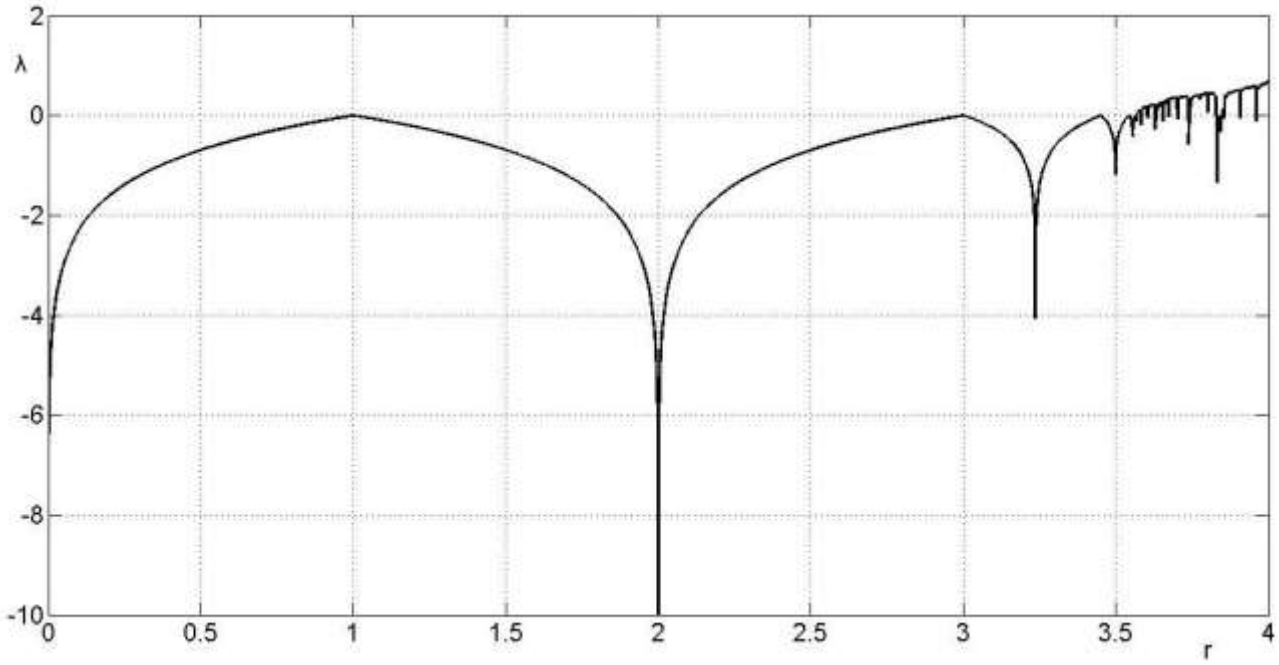

Figure №1. Lyapunov exponent for the logistic map.

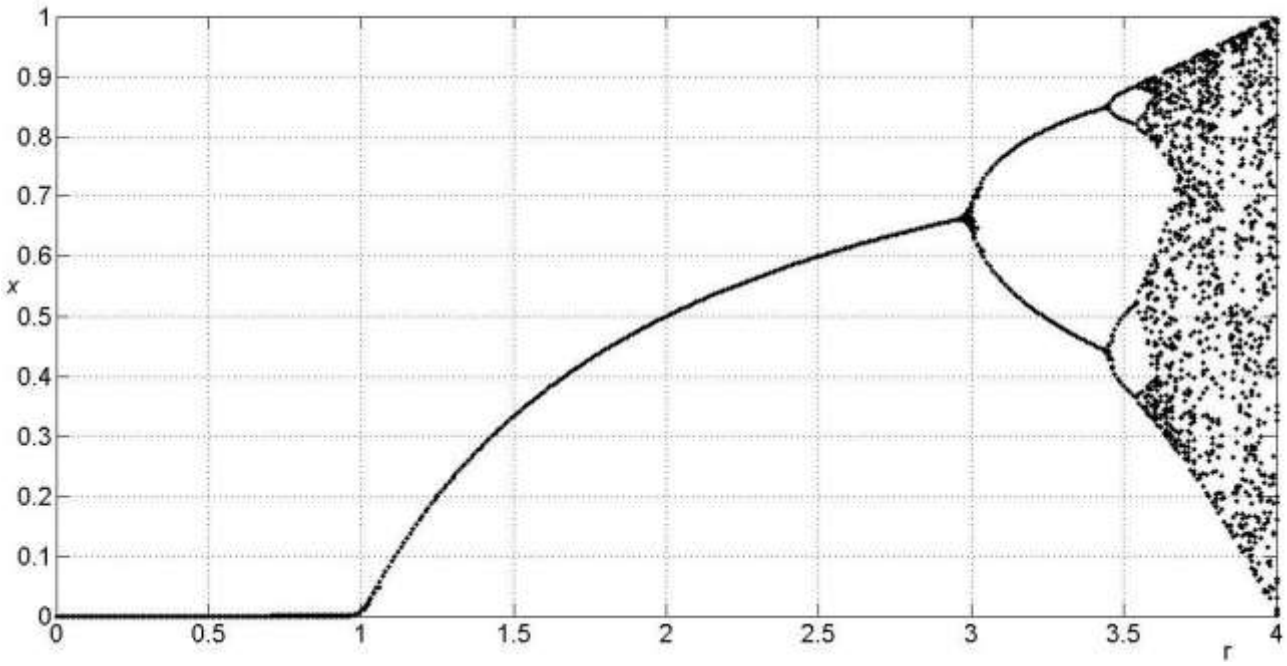

Figure №2. Bifurcation diagram for the logistic map.

Accuracy presentation of parameters x_n , r and x_{n+1} is the five decimal places. This allows the effective operation of multi-user system and eliminates the possibility of messages coincidence that received by different users.

Data transmission and addressing of users occurs through TCP / IP protocol with the statistical IP addressing and identifiable number of system resources on the server-side application (1). The storage of messages and keys of the system provided by an array of data on the server and client side. The algorithm of synchronization based on calculating and verifying checksums of iteration steps and the initial values of dynamic variables getting by the algorithm - CRC-32 - IEEE 802.3 (Figure №3). Checksums of initial values of dynamical variables is recording in the opening session message and compares with values backend (Figure №4). Checksum of iteration steps is attached in all messages and checked backend (2).

Figure №3. Block of initial synchronization

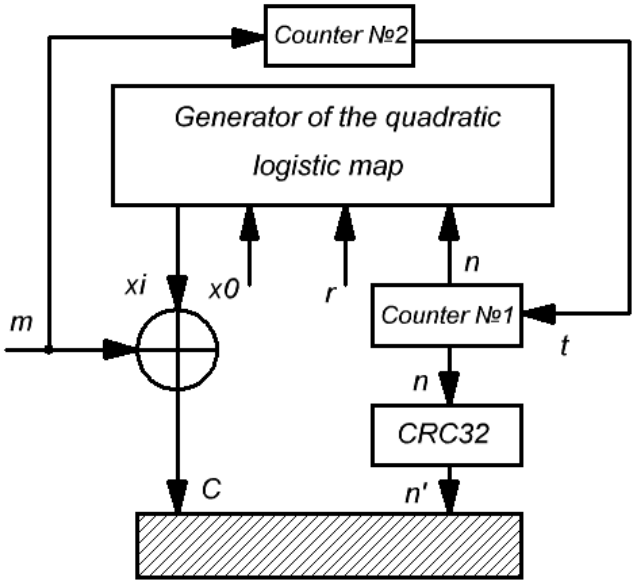

Figure №4. Block of encryption and synchronization

The beginning and the end of the exchange of information in the system is determined by the opening and closing sessions with commands between the server and the client part (Table №1).

Table №1. Commands of the transmission

№	Command	Description
1	GET (option) ADR (value) LGN (login)	Opening of the session between the client and server. If the previous session pending, the server will automatically open a new one. Transfer only from client part. (option) – checksum of initial values of dynamic variables. ADR –addressing command. (value) - the IP address of the sender. LGN – authorization command. (login) – username.
2	GETOUT	Sending information to the client-side.
3	GETIN	Sending information to the server part.
4	LGNX (login)	Recipient name (login). Messages without that command are recorded in the general array of messages.
5	ER1	Check error of GET command with the specified parameters. The server closes the session.
6	ER2	Synchronization Error . Inequality checksums. Both of the parts calculated previous iteration step.
7	ER3	Error of authentication command. The server closes the session.
8	ER4	Login of the sender is not in the array of users. The server closes the session.
9	MSG (message)	Beginning of the message.
10	CLR	Closing session (by the client part).
11	CLS	Closing session (by backend).
12	CS	Checksum of iteration steps

Algorithm of the transfer and synchronization system is represented by the following steps:

1) The system administrator specifies the parameters r and initial values of dynamic variables for each of the users on the client and server applications.

2) The client initializes the beginning of the session with command №1.

3) The server part generates a response using commands №2, №9, №12 and №4. If server part found an error, it sends command №5 - №8 instead of №9.

4) To make the transfer of information messages from the client side using command №3, №9, №12 and №4. If server part found an error, it sends command №5 - №8 instead of №9.

5) End of the session is initialized by client part (command №10) or by backend (command №11).

The proposed system can serve up to 20 users. The number of users depends on accurate representation of the parameters (in our case, to five decimal places). Increased accuracy of representation parameters enables more users but there will be a rise time data processing.

The proposed structure of the data set (Figure №5) and flags (information mantissa) of data array (Table №2) provides a performance of system. Array data is stored in plain text with UTF-8 coding and with separator commands that allows converting an array of data in CSV format and integrating with third-party software. Table of logging allows us to analyze the process of transmitting/receiving and view histories of using the system. Table of permission provides a distribution of rights among users, such as editing, storing and reading.

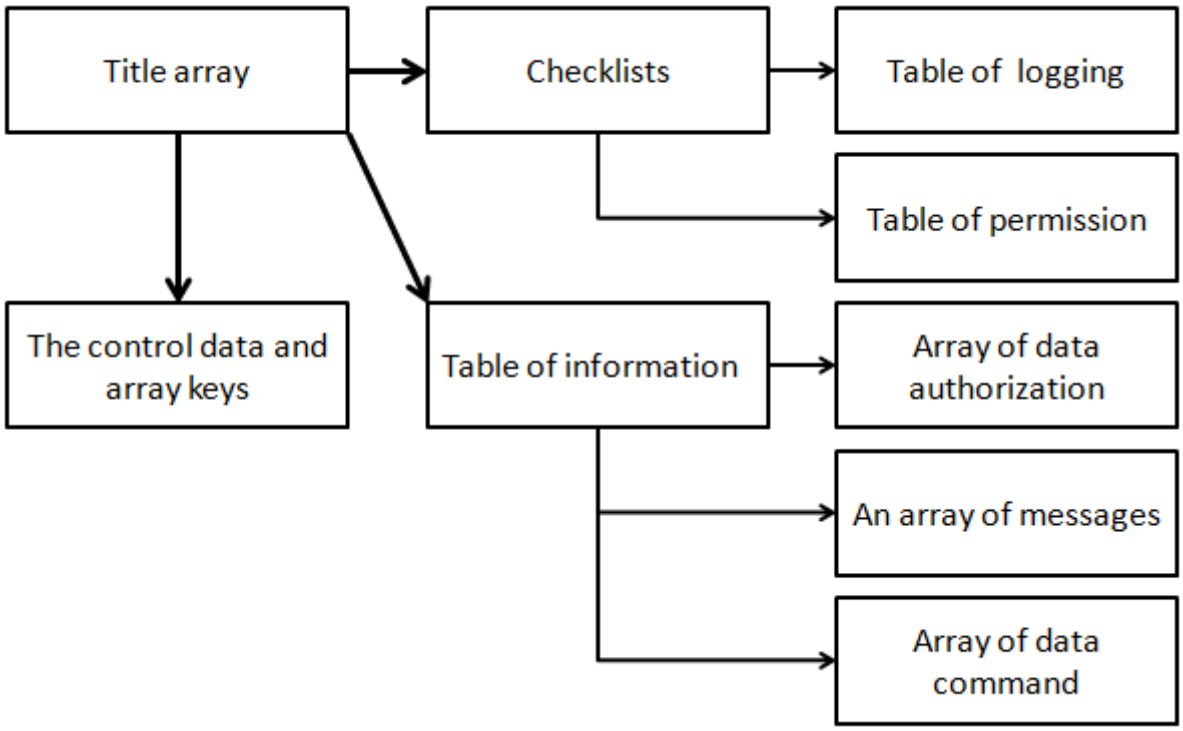

Figure №5. Structure of the data array

Table №2. Flags (information mantissa) of the data array

Flags (information mantissa)	**Description**
MSG	Flag of record message
KRCS	Flag of control data entry
LGN	Flag of record user login
PS	Flag of account user password
DT	Flag of record date

CMD	Flag of recording command
NODE(x) x = (A , B , C , D)	Flag of recording rights to change (A - all privileges, B - Read-only, C - record without reading, D- no rights to use)
;	Flag of separation
INFK	Flag of data arrays information

Conclusions

1) The proposed system allows the transfer of information to serve up to 20 users. It provided by a choice of presentation options accuracy to five decimal places.
2) Synchronization of logistic map generators performed by checksums of verification iteration step and the initial values of dynamic variables.
3) The structure of the implemented data array allows its conversion to CSV format and integration with third-party software.
4) The proposed structure of the system allows the implementation in scripting programming languages (Python) and launching on different operating systems.

References:

1. *Maufer Thomas A. IP Fundamentals. Prentice Hall; 1999; 3–34.*
2. *Philip Koopman. "32-Bit Cyclic Redundancy Codes for Internet Applications": The International Conference on Dependable Systems and Networks; 2002; 459–468.*
3. *Sprott Julien Clinton. Chaos and Time-Series Analysis. Oxford University Press; 2003; 9–21.*