Development of a model representing systems protected against research

M Styugin
Research fellow, Siberian State Aerospace University named after Academician M.F. Reshetnev, Krasnoyarsk, Russia
E-mail: styugin@gmail.com

Abstract. The research described herein presents a method of modeling information systems with dynamic parameters. The model includes elements with discrete internal states and agents, which can act in the system by modifying states of the system’s elements. The research has two objectives aimed at protection against research, which are as follows: inadmissibility for the system to transfer into some specific states and invisibility of actions taken by the agents. The present paper establishes stable secure states and proves algorithmic insolubility of Problem 1 in its general case..

1. Introduction
The task of information security lately is increasingly turning to the area of system protection against research by intruders at reconnaissance stage. The most well-known method in that area is the Moving Target Defense [1]. Over 150 different MTD methods [2] appeared within previous 5 years. However, problems related to evaluating efficiency of such methods still exist. Research studies [3, 4] are almost the only ones in this field. The present paper continues research started in [5] and [6]. The system will be represented as a unit, which consists of independent elements and agents. Each one of the agents is able only to see restricted system’s states..

2. Developing a functional model
Assume a system could be represented as a set of its states T. The system can transform from one state to another as a result of actions of agents.

Let us number agents in the system 1 to n. Each of the agents has a set of actions F_k. Each action represents a system mapping function from one state to another one:
\[
\forall F_i \in F_k \quad F_i : T_{i-1} \rightarrow T_i
\]

Hence, we can regard the composition of actions as a sequential transition from one state to another:
\[
F_i \circ ... \circ F_s : T_{i-1} \rightarrow T_s
\]
Thus actions or their combination are equal when initial and final system states are equal.

\[ F_i \circ \ldots \circ F_s : T_{i-1} \rightarrow T_s, \quad F_k : T_{k-1} \rightarrow T_k, \]

\[ F_i \circ \ldots \circ F_s = F_k \iff T_{i-1} = T_{k-1} \land T_s = T_k. \]

To show the information component of the conflict it shall be noted that each of the agents sees the system’s set of states differently and observes its changes differently. The reflexive structure or the awareness structure of agents in the system here could have been represented as a sequence of T sets in the conception about each other, e.g. as set forth in [7] or [8]. Although such representation of awareness is a most general one, but it is of little use from the practical point of view as in its case it is unclear where to obtain data for reflexive structure or awareness structure.

The present paper uses an assumption that not every system element is equally well observed by all agents. Hence, some particular system’s states are indiscernible for some agents. Such indiscernibility may be represented as the equivalence relation on set T.

\[ \forall k = 1..n \quad \exists R_k \in \mathbb{R} \]

Each system’s state is represented as an aggregate of independent elements

\[ T_i = (a^1_i, a^2_i, \ldots, a^n_i), a^1_i \in A^1, a^2_i \in A^2, \ldots, a^n_i \in A^n \]

Two system’s states \( T_i \) and \( T_j \) are indiscernible for agent \( k \) when

\[ T_i R_k T_j \text{ or } (a^1_i, a^2_i, \ldots, a^n_i) R_k (a^1_j, a^2_j, \ldots, a^n_j). \]

When in that case one of the elements is \( a^l_i \neq a^l_j \) hence element \( a^l \) is indiscernible for agent \( k \) in T system.

It is also evident that the agent cannot take actions resulting in changes of states in elements that are indiscernible for the agent. For instance the situation below is impossible for agent \( k \): 

\[ F_2 \in F_k \quad F_2 : T_i \rightarrow T_2, \quad T_i = (a^1_i, a^2_i)^T, T_2 = (a^1_2, a^2_2), \]

\[ (a^1_i, a^2_i) R_k (a^1_2, a^2_2) \land a^l_i \neq a^l_2. \]

In the example above the first system’s element changed its state as a result of agent \( k \)’s actions despite that its state is indiscernible for \( k \).

Thus it may be concluded that when a system changes its state due to actions taken by agent \( k \)
elements indiscernible for the agent remain unchanged.

\[ F_j \in F_k \quad F_j : T_i \rightarrow T_j. \]
\[ \forall a_i^s \in T_i, \forall a^s \in A^s, s = 1..n, \left( ..., a_i^s, ..., \right) R_k \left( ..., a^s, ..., \right) \Rightarrow a_i^s = a_j^s. \]

The above law may be denoted as the rule of indiscernible element conservation.

**Consequence 1.** An agent cannot transform a system into states indiscernible for that agent in accordance with its equivalence relations.

Undesirable states need to be introduced into system \( T \) so that the above model would allow formulation of the problem of system protection against research. Set of such states shall be noted as \( T^N \).

\[ T^N \subseteq T \]

All other states of the system may be regarded as acceptable.

\[ T^p = T \setminus T^N \]

May be represented by a tuple:

\[ \left\langle T, T^N, R, \{F\}, T_0 \right\rangle \]

Where \( T_0 \) is the current system state and \( \{F\} \) is a set of sets for all actions of agents in the system.

\[ \{F\} = \{F_1, F_2, \ldots, F_n\} \]

Based on the above model two different research protection tasks can be formulated:

1. The first task is related to preventing the system from transition to a state belonging to the unacceptable set \( T^N \). This problem can be solved from the research protection point of view only by correcting indiscernible system elements in relation to agents.

When state \( T^N \) is reachable in that system then it shall be denoted as a function’s value:

\[ \|\left\langle T, T^N, R, \{F\}, T_0 \right\rangle\| = 1 \]

Hence when \( T^N \) is not reachable, then
\[
\| \langle T, T^N, R, \{F\}, T_o \rangle \| = 0
\]

2. The second task is related to the difficulty of recognizing an agent which brought the system to the state belonging to \(T^N\) set or to any state in general. It is possible to detect such an agent with the present model by finding the chain of actions in its set which leads to an unacceptable state.

3. Solution to the first problem

First let us review a system with one agent:

\[ \langle T, T^N, R, F_1, T_o \rangle \]

**Theorem 1.** When for all \(T_j \in T^N\) \( \exists T_i \in T \setminus T^N : T_i R_i T_j \) and the condition
\[
\forall T_j \in T^N \, (a_0^s, ..., a_0^m) \subset T_j \text{ is fulfilled, where } (a_0^s, ..., a_0^m) \text{ are such that } \exists (a_j^s, ..., a_j^m) \subset T_j, a_j^s \neq a_j^i, ..., a_j^m \neq a_j^m \text{ then } \| \langle T, T^N, R, F_1, T_o \rangle \| = 0.
\]

Theorem 1 can be formulated differently as follows: when each system’s unacceptable state has a corresponding acceptable indiscernible state for the only agent then the system will never transit to it provided that the system has not been in that state initially in accordance with indiscernible elements.

A system with several agents may be reviewed in a similar way:

\[ \langle T, T^N, R, \{F\}, T_o \rangle \]

**Theorem 2.** When for all \(T_j \in T^N\) for all \(R_s \in R\) \( \exists T_i \in T \setminus T^N : T_i R_i T_j \) and condition
\[
\forall T_j \in T^N \, (a_0^s, ..., a_0^m) \subset T_j \text{ is fulfilled, where } (a_0^s, ..., a_0^m) = \bigcap_{s=1}^{n} (a_i^s, ..., a_i^h), \text{ and } (a_i^s, ..., a_i^h), \text{ are such that for every } s \text{ agent } \exists (a_i^s, ..., a_i^h) \subset T_i \text{ and } \exists (a_j^s, ..., a_j^h) \subset T_j : a_i^s \neq a_j^s, ..., a_i^h \neq a_j^h \text{ is fulfilled, then } \| \langle T, T^N, R, F_1, T_o \rangle \| = 0.
\]

In case the problem settings of Theorem 1 and 2 are not fulfilled it does not imply that the system may transit to an unacceptable state. When all potential actions of agents are reviewed in accordance with the \{F\} set then chains of actions equivalent to action \(F: T_o \to T_i, T_j \in T^N\). Let us consider the possibility to find the value of the characteristic function

\[
\| \langle T, T^N, R, F_1, T_o \rangle \| \to \{0, 1\}
\]

First it is required to find out whether there is an algorithm for defining the characteristic
function and hence whether the problem is possible to be solved algorithmically.

**Theorem 3.** Finding the algorithm for calculating the system’s characteristic function \( \langle T, T^N, R, F, T_0 \rangle \) is an undesirable problem.

The above theorems are examples of defining the Turing machine break points in a specific case. Given the restrictions applied to a specific task that list may be extended.

### 4. Solution of the second problem

The second problem is about the ability to correlate a transition in the system with an action of one of the agents. Here is a system of three agents and action F. Action F does not belong to a set of actions of any of the agents. However while a system transition took place hence action A may be represented as a composition of agents’ actions. Assume that

\[
F = F_1 \circ F_2 \circ F_3, \quad F_1 \in F_3, F_2 \in F_2, F_3 \in F_1.
\]

**Definition 1.** Trivial action is an action that cannot be presented as a composition of other actions which do not incorporate the action in question.

**Consequence 2.** An action that is not included into action sets of other system’s agents cannot be trivial.

**Definition 2.** A simple composition is a conception of an action that cannot be written with less elements. That is when there is conception \( F = F_4 \circ F_5 \) hence \( F = F_1 \circ F_2 \circ F_3 \) is not a simple composition.

Assume that in our case the chain \( F = F_1 \circ F_2 \circ F_3 \) is a simple one. It may be concluded that transition F in the system may be performed only by combined actions of agents 1, 2 and 3. Nevertheless it is not so, because each of the above actions may be not trivial. The whole composition of action is expanded as shown on Fig.1. Agent numbers are indicated above each action accordingly to an agent’s set of actions in which they are located.

That is an action \( F_1 \) may be presented as a composition of \( F_{11} \circ F_{12} \), where \( F_{11}, F_{12} \in F_1 \). Action \( F_{11} \) may be presented as a composition \( F_{111} \circ F_{112} \), where \( F_{111}, F_{112} \in F_1 \).
As a result action $F$ can be presented as a composition:

$$F = F_1 \circ F_2 \circ F_3.$$

All actions of the above composition are members of $F_1$ set. Hence agent 1 may have performed $F$ transition on its own.

When an action in a simple chain is replaced with a simple chain corresponding to it then it shall be referred to as a simple composition of a second order. When some action in the second chain is represented as a simple chain then it is a simple composition of third order. The above composition is a fourth order composition.

When no limitations are applied to representation of actions then in a general case all possible simple compositions in the system (functional decomposition) can be found only by enumeration.

Conclusion 3. The objective of functional decomposition has an exponential complexity from $|F|$.

When set $F$ is big enough it is impossible to perform functional decomposition hence it is impossible to find the agent which performed a transition in the system.

All this time the system was analyzed only from the outside with absolutely full awareness of its transitions and sets. In such conditions the task is always accomplished provided possession of unlimited computational power. From the system agents’ point of view the task cannot be accomplished due to indiscernibility of agents in some elements. Thus a transition in the system performed using functional decomposition may be unobservable for some agents.

Hence the system may appear in some state absolutely “unexpectedly” for agents. A most suitable definition for that problem is a term of “functional steganography”.

The subject of functional steganography is finding transitions in the system that are represented by functional decomposition of one agent’s actions and indiscernible for the other
system’s agents.
Hence there is a subset of actions from the area of functional steganography for each agent:
$$\forall i \in N \ \exists F_i^{(s)} \subseteq F_i.$$ That set might as well be empty. Transition from that set of actions means that other agents will associate that action (due to indiscernibility) with actions of the other system’s agents:
$$\forall F_k \in F_i^{(s)} \ \forall j \in N \ \exists F_j : F_k R F_j.$$ Rather interesting practical problems may be solved by defining such sets of actions for the system’s agents. An example may be a task of concealing actions of a security administrator in a system. The conclusion below may be useful for that task:

**Conclusion 4.** When one of the system’s agents has access to all system’s elements and the other agents have a set of indiscernible elements then a set of actions in the functional steganography of such agent is never empty.

A most simple implementation of that method can be performed with access control policies. Discretionary security policy can be an example where all objects are initially independent and indiscernibility may be regarded as no {read} right in the object access control list for subjects.

**Conclusion**

The present paper analyzed a mathematical model of an information system, which allows modeling insufficient awareness of agents about some of the system’s elements. The model can be used to find out, which system elements that are protected against research allow preserving the system in a secure state that is to avoid transition to an unacceptable state. The paper establishes initial states of an information system when it remains protected at all times. Two theorems were proven and based on the above accomplishment. Solutions obtained after resolving the problem provide an opportunity to design information systems protected against research with a proven security at the level of separate independent elements.

**References**

[1] Carvalho M., Ford R. Moving-target defenses for computer networks. // IEEE Security and Privacy. 2014. 12(2) pp. 73–76.

[2] JafarHaadiJafarian Q.D., Al-Shaer E. Openflow random host mutation: Transparent moving target defense using software-defined networking // Proceedings of the 1st Workshop on Hot Topics in Software Defined Networking (HotSDN). New York: ACM, 2012. pp. 127–132.

[3] Jajodia et al. Moving Target Defense. Creating Asymmetric Uncertainty for Cyber Threats. Series: Advances in Information Security. London: Springer, 2011. 184 p.

[4] Jajodia et al. Moving Target Defense II. Application of Game Theory and Adversarial Modeling. Series: Advances in Information Security. London: Springer, 2013. 203 p.

[5] Styugin M. Protection against system research // Cybernetics and Systems. 2014. 45(4). pp. 362–372.

[6] Styugin M. The New Method of Security Development for Web Services Based on Moving Target Defense (MTD) Technologies // Proceedings of the International Conference on Network Security and Communication Engineering (NSCE2014). Hong Kong: CRC Press, 2014. P. 130–136

[7] Lefebvre V. A. The Structure of Awareness. Beverly Hills: Sage, 1977. 177 p.

[8] Novikov D., Chkhartishvili A. Reflexion and Control: Mathematical Models. London: CRC Press, 2014. 298 p.