Industrial Control System Defense Decision-Making Method Based on Dynamic Attack-Defense Game

Shenyi Zhang\textsuperscript{1,a}

\textsuperscript{1}School of Computer Science and Technology, Beijing Institute of Technology, Beijing, China

\textsuperscript{a}e-mail: zsyzixi@gmail.com

Abstract. Industrial control systems are different from traditional information systems and require extremely high system availability. In the face of potential attack incidents, blind defense deployment may affect system performance and efficiency, which is counterproductive. For this reason, this paper proposes a defense decision-making method for industrial control systems based on dynamic attack-defense games. This method first analyzes the actual attack-defense confrontation scenarios in the industrial control system, and constructs a dynamic attack-defense game model based on this. After that, an industrial control system attack-defense game tree defined on this model is constructed, and the game tree is used to describe the specific attack-defense confrontation process. Then the backward induction method is used in the game tree to find the equilibrium path. The information on this path is of great reference value for defenders to choose a reasonable and efficient defense plan, maintain system stability and reduce defense costs. At the end of this paper, this method is applied to a real limestone powder preparation system for experiments. The experimental results verify the effectiveness of the method in actual scenarios.

1. Introduction

Industrial Control System (ICS) plays an important role in manufacturing and infrastructure construction, and therefore has different security features from traditional information systems, that is, more attention is paid to business continuity, production reliability and real-time system response. ICS managers are prone to make two kinds of mistakes. Either they overly consider the priority of the system's availability in production and operation and neglect the system's security protection, or they carry out reckless defense which leads to the reduction of system operation efficiency and the increase of economic cost. Therefore, when ICS security managers are deploying defenses, they need to predict attack methods and attack paths from the perspective of the attackers, in order to measure the expected benefits of different defense deployment methods and the negative impact on system availability, so as to select a low-cost, high-yield defense strategy.

In order to simulate the actual attack-defense confrontation process, researchers often use attack trees\cite{1}, attack graphs, Petri nets\cite{2}, and other methods to model attack and defense behaviors in the system. However, these modeling methods cannot reflect the game between the attacker and the defender that occurs on each device in the industrial control network. This paper uses the attack-defense game tree model, and describes the strategic choices of the attack and defense parties at each stage of the confrontation process through the connection relationship between the offensive and defensive nodes on the tree, which is specific and clear. Nguyen\cite{3} established a static game model of incomplete information to analyze the method of network defense strategy selection, but in the static game model...
it uses, both sides of the attack and defense are regarded as ignorant of each other’s strategy choices. Ge X[4] and Wang J[5] both use dynamic game models to show the interaction between the attack and defense parties in the choice of strategy, but the objects of their analysis are the ordinary information systems, which cannot reflect the attacker's penetration process in the industrial control system. This paper uses a dynamic game model suitable for ICSs, which can reflect that the attacker constantly adjusts his strategy and attack path according to the defender's strategic choice, while the defender also changes his defense strategy to limit the attacker's penetration.

When constructing the attacker’s strategy set, traditional methods mostly use atomic attacks or a combination of atomic attacks[6]. This method is more specific but does not directly link the attack behavior with the state of the equipment in the system, resulting in the lack of clarity and intuitiveness in their relationship. The game model constructed in this paper defines each attack strategy as a set of atomic attacks that enable the attacker to transfer from one device node to another device node in the system, thereby omitting some atomic attacks that have little to do with the penetration and making the attack path more explicit.

After constructing the game model and generating the attack-defense game tree, this paper presents an algorithm based on the reverse induction method to solve the game tree to obtain the attack and defense critical path. The extraction and analysis of the critical path can help identify key assets and key defense strategies, and help defenders to selectively allocate defense measures.

2. Attack-Defense Confrontation in Industrial control system

2.1. Analysis of Attack-Defense Confrontation Scenarios in Industrial Control Systems
ICS can be divided into four layers: field equipment layer, field control layer, process monitoring layer, and operation management layer. The operation management layer is usually the enterprise management network. In order to ensure the efficiency of financial management and human management in the company, it is often connected to the Internet and has a huge attack surface. Therefore, by default, it is strictly separated from the other three layers of the industrial control system through a firewall to achieve basic border defense. Nevertheless, the attacker's intrusion into the industrial control core system (used in this article to refer to the three-layer system consisting of the field equipment layer, the field control layer and the process monitoring layer) still has multiple attack surfaces such as malicious code implantation and unsafe remote connections.

Consider the industrial control core system as a graph formed by each ICS device in the network as a node[7], after invading the industrial control core system, the attacker will start from the invaded nodes and use various vulnerabilities in the system to gain control of the devices in the system one by one. In this process, in order to maximize their own interests, the attacker will proceed along the path of smaller costs and greater benefits. The defender needs to take targeted and selective defenses on each side of the path by predicting the attack path of the attacker. By that he could directly cut off the attacker's attack path on the key assets of the system, or limit the attacker's target choice and gradually guide it to the low-asset node, or abandon the defense of low-risk edges to save defense costs. The interests of the attacker and the defender conflict with each other and their actions are mutually restricted. They are strategically dependent, and the relationship of them belongs to a non-cooperative game.

2.2. Dynamic Attack-Defense Game in Industrial Control System
The attack-defense confrontation models in traditional information systems are often constructed with system status and permissions as nodes. The defense of ICSs needs to take equipment as the core and formulate defense strategies around the accessibility of key industrial control equipment assets by attackers. Defenders can take targeted defenses such as installing patches and adopting highly encrypted protocols on specific devices, thereby completely blocking or changing the attack path of the attacker. Considering the above, this paper proposes an attack-defense game tree model for ICSs based on the connection between industrial equipment. The following uses a simple attack-defense confrontation scenario in the ICS as an example for illustration.
Figure 1. Simple attack-defense confrontation scenario

In Figure 1, A, B, and C respectively represent three vulnerable devices in the ICS network. $a_1$ and $a_2$ are both attack strategies to invade node B from node A. $a_3$ is the attack strategy to invade node C from node A. $a_i(d_i)$ indicates that defense method $d_i$ can block attack $a_i$. Suppose the defender predicts that the attacker will reach node A when deploying defense measures, the defense methods of the defender at this time are $d_1$, $d_2$, and $d_3$. Then the defender’s strategy set is a combination: $S^d = \{0, d_1, d_2, d_3, (d_1, d_2), (d_2, d_3), (d_1, d_3), (d_1, d_2, d_3)\}$, where 0 means that no method will be taken. When the attacker reaches node A, the defender has deployed defense measures. After observing these defense measures, the attacker will abandon the use of blocked attack methods and adopt other strategies. Assuming that the attacker does not attack multiple devices at the same time, that is, only attacks one of B or C, then there is a game tree as shown in Figure 2.

Figure 2. Simple attack-defense game tree

In Figure 2, the white nodes and black nodes in the figure represent the decision nodes of the defender and the attacker, respectively. For simplification, the figure omits the attacker’s choice of "no action" strategy on each decision node except for the leftmost node. When the attacker succeeds in the attack, the attacker reaches a new node, i.e. node B or C in the figure, and the game will continue according to the above steps. When the attacker chooses "no action" strategy, the attacker is deemed to have withdrawn from the game and the game tree generates a leaf node below this choice. The previous choices of the attacker and the defender constitute a path from the root of the game tree to the leaf node. For example, the leftmost node in the figure is a leaf node and the path related to it is $(d_1, d_2, d_3, 0)$. By counting the defense cost, attack cost, asset value and other information on this path, the total utility value of the attacker and defender corresponding to this path can be obtained.

From the above analysis, it can be seen that the actions of the attacker and the defender are sequential and alternate, the actions of the previous step determine the optional strategies for the next step, which
is a typical dynamic game. In the complete attack-defense confrontation scenario of an ICS, at first, the defender understands the network topology and vulnerabilities of his own system, and understands what attack methods the attacker will use to exploit the vulnerabilities. Secondly, the information that the defender knows can be obtained by the attacker through information collection methods such as scanning and Internet searching, then the defense method information can be inferred. Now suppose that the attacker’s starting position in the industrial control core system is the common information of both parties, then the game between the two parties can be regarded as a complete information game. Therefore, the attack-defense confrontation in the ICS can be represented by a complete information dynamic game model. The model is defined as a four-tuple $G = (H, S, T, U)$.

$H = (H_A, H_D)$ is the set of players in the game, $H_A$ represents attacker, $H_D$ represents defender. $S = (S^a, S^d)$ represents a collection of attack strategies and defense methods. $S^a = \{s_1^a, s_2^a, \ldots, s_n^a\}$ is the set of attack strategies. For any $s_i^a$ in the set, $s_i^a = (n_i, t_i, c_a, Z)$ = (Starting device, target device, attack cost, attack condition). Each attack strategy is not necessarily an atomic attack, but it may also be the smallest set of attack processes on the starting device that enables the attacker to obtain the control authority of the target device (that is, no redundant atomic attack). The cost of the strategy refers to the sum of the atomic attack costs included in this attack process. $S^d = \{s_1^d, s_2^d, \ldots, s_m^d\}$ is the set of defense methods. For any $s_i^d$ in the set, $s_i^d = (K, c_d)$ where $K$ is a set containing all attack strategies that can be prevented by this defense method. $c_d$ represents the defense cost of the defense method.

$T$ represents the attack-defense game tree defined in this game. The attack-defense game tree is the embodiment of the game process.

$U = (U_A, U_D)$ represents the utilities of the two players. Attack utility $U_A$ is the net income that the attacker obtains in a complete attack-defense confrontation (the confrontation is reflected in the game tree as a path from the root node to the leaf node). That is, the difference between the attack return and the attack cost of all strategies adopted by the attacker in this confrontation. Because by forcing the defender to adopt a high-cost defense strategy, sacrificing the availability of the system, or making the defender pay a financial price, the attacker can also gain benefits, the attack return not only includes the asset value of the compromised device, but also the defense cost of the defender. On the contrary, a defender cannot benefit from the attacker’s price. So defense utility $U_D$ is simply the sum of equipment assets lost in the confrontation and the cost of adopting defense strategies. Suppose an attack-defense confrontation is represented by path $L$:

$$U_A = \sum_{i \in L} (Asset_i + DC_i) - \sum_{i \in L} AC_i$$

(1)

$$U_D = -\sum_{i \in L} (Asset_i + DC_i)$$

(2)

### 3. Industrial Control System Attack-Defense Game Tree

ICS attack-defense game tree is represented by a three-tuple $T = (N, R, Depth)$. $R$ represents the root node of the tree. At the beginning of the whole attack-defense confrontation process, because the attacker is deemed to have controlled a device in the industrial control core system, the defender will take defense measures against this device, and the attacker will then adopt strategies to attack other devices from this device. Therefore, the root node is the defense node, and its corresponding device is the beginning of all possible attack paths. $Depth$ is the depth of a tree, i.e. the maximum level of leaf nodes. $N$ represents the node set of the tree $N = \{n_1, n_2, \ldots, n_l\}$ where $l$ is the total number of nodes. The nodes in $N$ are divided into two types, attack nodes and defense nodes. The properties of the two types of nodes are different, as shown in Table 1.
Table 1. Description Of Node Properties

| Property              | Description                                                                 |
|-----------------------|-----------------------------------------------------------------------------|
|                       | **Attack Node**                                                             | **Defense Node**                                                          |
| Type                  | Type=0 means this node is an attack node                                     | Type=1 means this node is a defense node                                   |
|                       |                                                                             | The corresponding device of the defense node is the target node of the     |
|                       |                                                                             | strategy that the previous node (an attack node) adopted                  |
| Corresponding Device  | Not defined                                                                 |                                                                             |
|                       |                                                                             |                                                                             |
| Candidate Attack      | All attack strategies available for this node                               | Not defined                                                                |
| Strategies            |                                                                             |                                                                             |
| Candidate Defense     | Not defined                                                                 | All defense methods available for this node                                |
| Methods               |                                                                             |                                                                             |
| Candidate Defense     | Not defined                                                                 | A set of all combinations of defense methods available for this node      |
| Strategies            |                                                                             |                                                                             |
| Terminal Node Flag    | Fin=1 means this node is a terminal node; Fin=0 means this node is not a     | Same as left                                                               |
|                       | terminal node                                                               |                                                                             |
|                       | The number of layers where the node is located on the game tree (the level of |                                                                             |
|                       | the root node is 0)                                                         |                                                                             |

The following are explanations of some special situations that are not involved in the simple attack-defense confrontation scenarios in Chapter 2.

The attacker will not attack the compromised device, because the attacker will not gain benefits but will only increase the cost.

In the same path, the attack node below will inherit the attack strategy that the attack node above did not choose.

For example, in Figure 3, the left side is a part of a game tree (black nodes represent attack nodes, and white nodes represent defense nodes), the right side is the relationship of the three industrial control equipment A, B, and C involved in the game tree. In the first layer of the game tree, the attacker has two attack strategies at A: \( a_1 \) and \( a_2 \), and the attacker chooses \( a_1 \) to reach B. In this path, \( a_2 \) becomes the attacker's unselected strategy. This strategy will be inherited to the set of candidate attack strategies at the lower attack node for selection. For example, in the third layer of the game tree, the attack node chooses the attack strategy \( a_2 \) (the corresponding scenario is that the attacker returns from B to A to launch an attack). After \( a_2 \) is selected, \( a_2 \) will no longer appear in the set of candidate attack strategies of the attack node below. In contrast, the defender can only take its defense measures against the current corresponding device at the defense node, and cannot defend against attacks launched by the attacker.
on other devices in the current path. Because the defender can be regarded as having made choices on those devices, those choices cannot be changed afterwards.

One defense method may prevent multiple attack strategies. For example, in a real scenario, a properly configured firewall can cause multiple attack strategies to fail. In order to reflect this in the model, the defense methods taken at the top of a path will be recorded, the defense nodes below will not take repeated defense methods, and the attack nodes below will also remove the strategies that have been determined to be blocked from the set of candidate attack strategies.

Atomic attacks included in some attack strategies need to be executed on multiple devices and therefore require control of multiple devices. In order to reflect this, the attribute of "attack conditions" is defined for the attack strategy, which is represented by the set $Z$. $Z$ contains the equipment that must be controlled to implement the strategy. After all these devices are included in the path of the attack node, the strategy will be added to the set of candidate attack strategies.

4. Extraction of Critical Defense Path of Industrial Control System Based on Reverse Induction

According to the previous analysis, the game model $G$ represented by the attack-defense game tree is a dynamic game of complete information. Since the two parties take turns in the game, and the lower participant accurately understands the actions of the upper participant when taking the action, and the game has a finite number of rounds, the game belongs to finite perfect information game. Zermelo[8] pointed out that a finite perfect information game has a purely strategic Nash equilibrium. So when both parties are completely rational[9], there is a path from the root node to the leaf node in the game tree, and through this path, both attacker and defender get a result that is most beneficial to them. The equipment in this path represents the key asset in the ICS. The defense deployment and attack strategy selection on the path have important reference value for the defender.

In order to find this path, the reverse induction method can be used to continuously compare the attack and defense utilities of each path all the way up from the bottom of the game tree. The principle of reverse induction is not complicated. Since the game behavior occurs sequentially, the rational game player who acts in the previous stage will inevitably consider how the latter game player will act in the later stage. A player can make a clear choice directly only if they choose in the last stage of the game because at that time he is no longer affected by any subsequent stages. Once the choice of the player in the later stage is determined, the behavior of the player in the previous stage is also determined accordingly. By continuously removing the bad choices in the sub-game from the bottom of the tree upwards, the last remaining path is the equilibrium path, and the strategy choices of both parties on this path reach the subgame perfect Nash equilibrium[10]. The reverse induction algorithm in attack-defense game tree is given as follows:

**Algorithm 1 Reverse Induction Algorithm of Finding equilibrium Path in Attack-Defense Game Tree**

**Input:** attack-defense game tree

**Output:** equilibrium path $L$, the attack strategy set on the equilibrium path $S_a$, the defense strategy set on the equilibrium path $S_d$, the attack and defense utilities corresponding to the equilibrium path $(U_a, U_d)$

**Initialize $T$**

for ($i = Depth, i > 0, i--$)

for node s.t. $node.fin == 1$ and $node.D == i$

node.father.fin = 1

if $i % 2 == 0$ // If the level is even, the node is a defense node, and its parent node is an attack node

if $node.father.Us < node.Ua$ or ($node.father.Us = node.Ua$ and $node.father.Ud > node.Ud$)/ If the attack utilities of the two paths of two nodes are the same, the attacker will choose the one with lower defense utility

$node.father.Us = node.Ua$

$node.father.Ud = node.Ud$

if $i % 2 == 1$ //If the level is odd, the node is an attack node, and its parent node is a defense node
if node.father.U_a == 0
    node.father.U_a = node.U_a
    node.father.U_d = node.U_d
else if node.father.U_a < node.U_d
    node.father.U_a = node.U_a
    node.father.U_d = node.U_d
else if node.father.U_a = node.U_d and node.father.U_d > node.U_a
    // If the defense utilities of the two paths of two nodes are the same, the defender will choose the one with lower attack utility
    node.father.U_a = node.U_a
    node.father.U_d = node.U_d
print (root.U_a, root.U_d) // The attack and defense utilities in equilibrium obtained by the reverse induction method is saved at the root node
for node s.t. node is leaf // If a node is a leaf node, judge whether it is the leaf node of the equilibrium path
    if node.U_a = root.U_a and node.U_d = root.U_d
        print node.L
        print S_a
        print S_d

5. Case Analysis
In order to illustrate the effectiveness of the previously proposed dynamic attack-defense game model and related algorithms, the security analysis of the real limestone powder preparation system is carried out. The network topology of its core control system is shown in Figure 4. The core system includes 5 vulnerable devices. Assuming that an attacker has controlled the engineering station through phishing attack, and will use this as a starting point to penetrate the system. The vulnerability information on each device and the attack strategy that the attacker may adopt to penetrate between each device can be abstractly represented by Figure 4 and Table 2. For each attack strategy that the attacker can take, the defender can use the defense methods in Table 3 to stop it. Information on equipment asset values, attack costs, and defense costs are obtained through industry historical data and expert opinions, which are shown in Table 2 and Table 3.

Figure 4. Topology of example system
Table 2. Information Of Equipment Vulnerability

| Equipment | Description       | Asset | Vulnerability             | Corresponding Attack |
|-----------|-------------------|-------|--------------------------|----------------------|
| A         | Engineer station  | Not defined | Compromised            | Not defined         |
| B         | Operator station  | 10    | Cve-2011-4876            | $a$ (5)             |
| C         | SCADA Server      | 20    | Cve-2011-2214, Cve-2011-1566 | $a_2$ (8), $a_3$ (10) |
| D         | PLC1              | 50    | Cve-2018-7761            | $a_4$ (20)         |
| E         | PLC2              | 40    | Cve-2018-7761            | $a_4$ (20)         |

Table 3. Description Of Defense Method

| Defense Method | Defense Cost | Description                      | Corresponding Attacks |
|----------------|--------------|----------------------------------|-----------------------|
| D1             | 8            | Increase access control          | $a$                   |
| D2             | 60           | Update 7T IGSS                   | $a_2$, $a_3$          |
| D3             | 10           | Close TCP port 12397             | $a_3$                 |
| D4             | 50           | Disable HTTP service             | $a_4$, $a_5$          |

Figure 5. Game tree of the attack-defense confrontation

Based on the information above, use the method of this paper to construct an attack-defense game tree to describe the game process, as shown in Figure 5. The equipment asset value, attack cost, and defense cost are calculated using (1) and (2), and the attack utility and defense utility represented by each complete path in the game tree are obtained.

By applying the reverse induction method in Chapter 4 to the game tree, the equilibrium path can be obtained as $d_1, d_2 \rightarrow o$, represented by a green path in the figure. The path means that the defender not only protects device B through access control, but also eliminates the vulnerability of C by installing patches, thereby blocking all attack methods of the attacker, making it impossible to reach D and E. This shows that from the perspective of the defender, although the installation of patches will cause the suspension of industrial production and incur greater defense costs, it is necessary to block the attack path in order to prevent attackers from hacking high-asset devices D and E. In contrast, taking the red path in the figure as an example, this attack path is $o \rightarrow a_1 \rightarrow d_1 \rightarrow a_2 \rightarrow o \rightarrow a_4 \rightarrow o \rightarrow a_5$, which
means the attacker compromised all devices in the system. At this time, the defense utility is -130, which is completely unacceptable to the defender.

But if it is only to defend high-asset equipment, another possible option for the defender is to adopt defense method $d^1$ instead of high-cost defense method $d^2$. But in this case, the attack and defense utility of a path $o \rightarrow a_1 \rightarrow o \rightarrow a_2 \rightarrow o$ that better reflects the expectations of both parties (indicated by the blue path) is (67, -80). Although the number is close to the utility value of the equilibrium path (68, -68), the path is still filtered out in the third layer of the tree indicated by the dashed line in the figure. This reflects the characteristics of both attack and defense parties pursuing the maximization of benefits. It also reflects from the side that although device C has low asset value and high defense costs, a manager cannot easily give up protection to this equipment when deploying defensive measures.

6. Conclusions

This paper constructs a dynamic attack-defense game model for the attack-defense confrontation that occurs in the ICS, and proposes a method to describe the game process using an attack-defense game tree. A reverse induction algorithm in the game tree is given which can give clear results by solving the equilibrium path, thereby helping ICS defenders deploy defensive measures. The experimental results show that the method in this paper can provide a reference for the managers of ICSs to carry out efficient defense. However, when the network topology of the system is relatively complex, the generation of the game tree is prone to state explosions. So a suitable simplified method for the game tree model proposed in the paper will be found.

Acknowledgements

Many thanks to my supervisor Professor Yan. This study is supported by The National Key Research and Development Program of China(2016YFC1000301).

References
[1] Byres, E. J., M. Franz, and D. Miller. (2004) The use of attack trees in assessing vulnerabilities in SCADA systems. In: ieee conf.international infrastructure survivability workshop. institute for electrical & electronics engineers.
[2] Bouchti, A. E., and Haqiq. (2013) Modeling cyber-attack for SCADA systems using CoPNet approach. In: International Conference on Complex Systems IEE.
[3] Nguyen, K. C., T. Alpcan, and T. Basar. (2009) Security Games with Incomplete Information. In: Proceedings of IEEE International Conference on Communications, ICC 2009, Dresden, Germany.
[4] Ge X, Zhou T, Zang Y, Zhu J. (2020) A Dynamic Defense Strategy Selection Method Based on Stackelberg Security Game. Computer Engineering and Applications, v.56. No.960.17:134-140.
[5] Wang J, Zhang H, Na W, et al. (2016) Defense strategy selection based on signaling game. In: International Conference on Cyberspace Technology IET.
[6] Jiang W, Fang B, Tian Z, et al. (2010) Research on Defense Strategies Selection Based on Attack-Defense Stochastic Game Model. Journal of Computer Research & Development.
[7] Li, and P. Shuang. (2014) Active Defense Technology in Network Security Based on Non-Cooperative Dynamic Game Theory. Applied Mechanics and Materials. , 687-691:1892-1895.
[8] Schwalbe, U, and P. Walker. (2001) Zermelo and the Early History of Game Theory. Games and Economic Behavior., 34.1:123-137.
[9] Zhang W. (2012) Game theory and information economics. GezhiPress, Shanghai.
[10] RJ Aumann. (1995) Backward induction and common knowledge of rationality. Games and Economic Behavior., 8.1 : 6-19