MedCrave
Step into the World of Research

Aeronautics and Aerospace Open Access Journal

Technical Paper

Open Access

CrossMark

A trust verification strategy for autonomous control system in launch site

Abstract

The control system at the launch site has gradually started providing unattended autonomous control during the flight mission. The autonomous control system is very important to autonomous control for ground facilities and equipment in space launch. Even if the autonomous control system has been tested, there is still no very effective method to ensure the trusted operation of the autonomous system. We propose a strategy for the trusted operation and verification of autonomous control system. Based on the state transition model of the control system, the trust verification of autonomous control system ensures the trusted operation of autonomous control. This verification is set up in the intelligent support system of the host computer, which makes lightweight for verification calculation in the control system. Trust verification strategy is the trusted stamp and the verification protocol. The state transferring of each module inside each controller is stamped by the stamping module embedded in PLC. The control output set includes the state stamps and is sent to the intelligent support system. The control verification module of the intelligent support system verifies the correction of autonomous control. The trusted state transferring is used to ensure the credibility of the autonomous operation. The paper analyzes the strategy's security under environmental impact as well as independent and joint attack. The strategy is applied to a prototype system. It illustrates the feasibility of the strategy at launch site.

Keywords: launch site, autonomous control system, trust verification, trusted stamp, verification protocol

Volume 3Issue 3 - 2019

Litian Xiao, Mengyuan Li, Kewen Hou, Fei Wang, Yuliang Li
Beijing Special Engineering Design and Research Institute, China

Correspondence: Litian Xiao, Beijing Special Engineering Design and Research Institute, Beijing 10028, China, Tel 86-10-56253387, Email xiao_litian@sina.com

Received: August 21, 2019 | **Published:** September 13, 2019

Introduction

At present, the autonomous control of the control system is more and more widely used in the ground facilities of the launch site. Programmable Logic Controller is a kind of embedded system in the autonomous control system. The controller's trusted operation is vital to this kind of safety-critical applications, especially in space launch missions. Conceptually, the trusted goal of the control system is to propose a running entity that can perform a special action surpassed the preset security rules, and how to verify the entity. Therefore, we need to first ensure that the software and hardware systems can be quantified and verified by calculation. The critical core of a trusted system is trusted computing. At present, the trusted control system platform is mainly considered the security issues from the perspective of security architecture. The security of the user's operating environment is ensured by the passive defense (i.e. security patching). We are committed to ensuring trusted operation and verifiability for autonomous control systems at the space launch sites. The paper analyzes the state change and transmitting of the control system, and proposes a trust verification strategy for the autonomous control system in launch site. Based on the proposed strategy, the security mechanism and the processes are set up by the critical elements for verifying the trusted autonomous operation. Once the trusted problem is verified, the problem is submitted to the fault handling module to analyze and deal with the autonomous control system. So the strategy ensures the operation reliability and security of the whole autonomous control system.

Related work

Control system architecture

The control objects of the ground facilities and the equipment control system at the launch site include pendulum bars, launch tower platforms, propellant refueling stations, gas supplies, air conditioners, fire protection devices, cranes, aim windows, gas alarms, moving towers, etc. A general control system for the launch site can be described by a state-transfer model.[1,2]

The control system receives and processes the various sensor signals from the controlled devices. These sensor signals (Sig_{sen}) include analog signals (AI) and digital signals (DI), e.g., the signals from coders (Cod), location detectors (Loc), operation inductors (Opt), and moving sensors (Mov). They are converted and stored in the data input port (PI) by analog-to-digital (A/D) and digital-to-analog (D/A) convertors. The sensor signal set is

$$Sig_{sen} = \{AI_{Cod}, AI_{Loc}, AI_{Opt}, AI_{Mov}\ldots, DI_{Cod}, DI_{Loc}, DI_{Opt}, DI_{Mov}\ldots\}; PI \subseteq Sig_{sen}.$$

According to the control logic, initial configuration, task, and interrupt, the control program processes and responds to the port data. Then, it updates the system configuration (V) and provides the corresponding control data to the data output port. After A/D or D/A conversion, the control data are converted to analog signals or digital drive ones (AO or DO) for moving the drive mechanisms such as relays, valves, transmissions, etc. Finally, real-time control is achieved, i.e.,

$$Sig_{drv} = \{AO_{Rly}, AOV_{alv}, AO_{Trans}, \ldots, DO_{Rly}, DO_{Valv}, DO_{Trans},\ldots\}; PO \subseteq Sig_{drv}.$$

The control system is constructed using components shown in Figure 1. The figure describes the work pattern and system model of the control system. A control system can be composed of multiple controllers, and the controller's system features are decided by $\{PI, V, PO\}$, i.e., the controller's action is controlled by its input, output, and configuration.[3]

The control system at the launch site has gradually started providing unattended autonomous control during the flight mission. The control

system for ground equipment plays an important role in ensuring accurate and reliable control of each system, and in completing the test preparation, propellant refueling, and launching. Because of the limited resources of the field control systems, verifying the accuracy of autonomous control is not completely possible, which may result in security risks.

Intelligent support must make system-performance forecasts when the critical equipment control performance has changed. It must have simple and convenient maintenance procedures and must ensure and must ensure that the system is always in a good condition. In the event of a sudden control fault, it must make a rapid and accurate diagnosis, and based on the location, provide countermeasures on how to handle the situation appropriately. It must maintain control safety and reliability during the mission. These requirements can be met by constructing an intelligent support architecture with remote monitoring, control verification, fault diagnosis, and prediction. Intelligent support needs to provide work capabilities under multi-mission and multi-field scenarios. The designed remote intelligent support architecture for the ground equipment at the launch site is shown in Figure 2. The architecture is divided into two levels: field level and launch-site level.

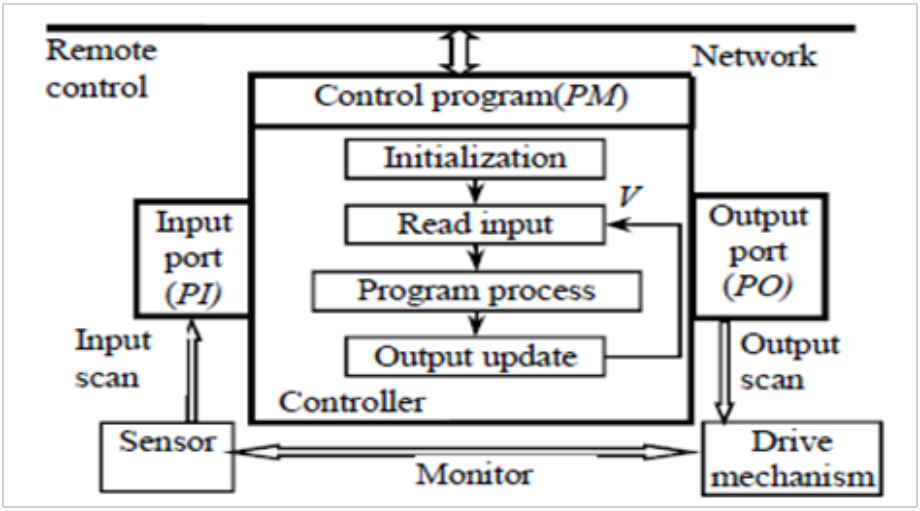

Figure 1 Work pattern and system model of control system.

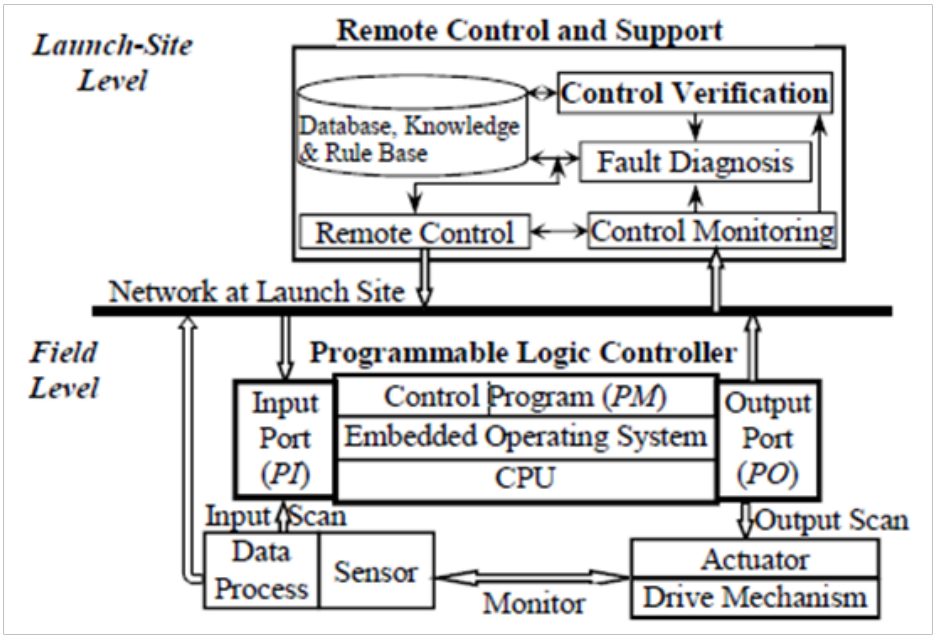

Figure 2 Remote intelligent support architecture.

Field level

The control system for ground equipment is distributed at the launch site. Usually, the control system completes the control tasks independently, according to the task flow. Operators can operate the system through close control. In the case of intelligent autonomous control, the field will consume more computing resources. Field verification of whether autonomous control is correct or not requires the deployment of additional resources. These will cause field-resource strains and inconveniences in terms of the control equipment

Citation: Xiao LT, Li MY, Hou KW, et al. A trust verification strategy for autonomous control system in launch site. *Aeron Aero Open Access J.* 2019;3(3):127–132. DOI: 10.15406/aaoaj.2019.03.00090

layout and maintenance. Field control systems become more complex. Therefore, the front-end of the automatic control system is set up to be lightweight, to simplify the field control system.

Launch-site level

This level is located at the command center of the launch site. Control verification verifies whether the front

Autonomous control is operating correctly or not. In the case of faults, the fault-diagnosis system automatically performs the diagnosis. Operators, technicians, and supervisors monitor the performance state of ground equipment in real-time, who can maintain system equipment and control programs. The level performs remote control in the case of abnormal situations and coordinates with remote experts for technical support and guidance. The hierarchical remote intelligent support system is briefly described as follows:

a. The field level performs autonomous control.

b. The launch-site level is for autonomous control verification and fault diagnosis.

c. The long-range level provides intelligent technical support for front-end autonomous control faults and maintenance.

The intelligent support processes involved in the control task are shown in Figure 3. The architecture key is control verification.

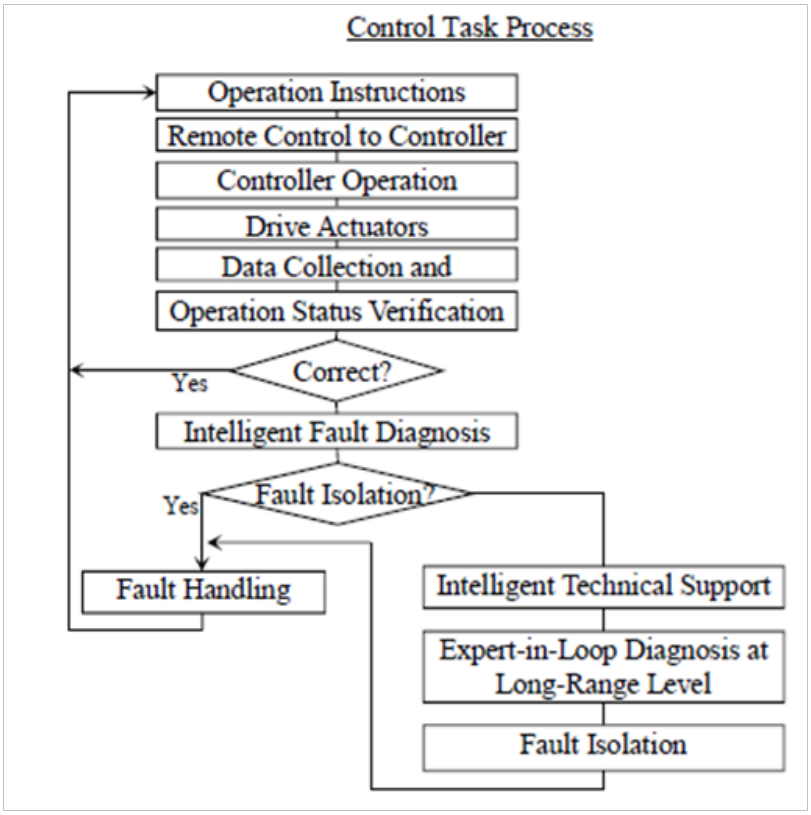

Figure 3 Intelligent support processes in control task.

Trusted research of control system

Now, there are a few related papers and reports about the trusted research of control system. The research of the related trusted system is mainly focused on the network, trusted computing and related software and hardware.[4–11] For the trusted guarantee of the complex software system under the network environment, the main researchers are that establishing a new software technology hierarchy. In order to adopt the hierarchy transformation, they study software object, quality target, construction method, and operation support. The quality model of internet ware is established with the quality core. The trusted technology system is constructed by quality assurance. In the research and development of trusted computing, some scholars believe that trusted computing cannot solve all information security problems. Trusted computing technology is combined with other information security technologies in the trusted computing field. They hope that they effectively solve information security problems.

In the research of software trusted evolution management, it is considered that the nature of trusted evolution is a software modification process driven by user expectation deviation. It makes that the software system is gradually from a simple to complex and from the preliminary use to mature and credible process. In the evolution view, it studies how to improve the convergence speed and effect of software trusted evolution.

The researchers on the architecture and key technologies of trusted networks propose a possible control model for evaluating the enabling technologies of trusted networks. They analyze the credibility of the user behavior in the network access process corresponding to the control rights model. In the progress of trusted computing technology hardware, the software on the computing platform is measured by enhancing the hardware architecture of the existing open platform.

Citation: Xiao LT, Li MY, Hou KW, et al. A trust verification strategy for autonomous control system in launch site. *Aeron Aero Open Access J.* 2019;3(3):127–132. DOI: 10.15406/aaoaj.2019.03.00090

The software on the platform can be ensured to operate as expected. In the control system field, the technical features of the trusted control system are integrated existing defensive measures. The researchers utilize the linkage mechanism between the inside firewall of control systems, the intrusion detection system and the trusted connection server. It can improve the overall defense and security capability of control systems,[12,13] but it doesn't have the trusted verification and cannot ensure the trusted operation.

The credible risks are very high in actual applications at the aerospace launch site. For example, the control system of propellant injection is a typical system. It requires monitoring of temperature, pressure, and flow velocity, valve opening size, etc. It involves each valve action sequences and time as well as pipeline pressures while controlling valve opening or opening size. If the control information is incorrect, the propellant injection system may cause leakage or tube burst and freezing. Its consequences are catastrophic. Certainly, measures have generally been taken for these keys in the control system. Some have adopted redundant or hot backup methods. Some have added monitoring points and fault handling software. Others are strict with the test inspection and control the identity of software users. But there is no trusted verification in control system, whether the state transmission can be trusted exists the risk on the information source and host as well as terminal. Therefore, the untrusted verified system exists risks on safe and reliable. Trusted computing or credibility verification has become the main issue in the field of information security. However, correlated trusted research lags behind the application needs in actual control systems. It still lacks typical solutions.

Trusted Elements in Control System

Based on the control system model,[3] ensuring the trusted state transferring in the controller is the key to ensure the credibility of the autonomous operation.

Set $M = < V, v_0, PI, PO, R_{PI}, PM>$, where V is the finite-state set of nonempty configurations inside the controller, PI is the finite set of non-empty inputs, PO is the finite set of nonempty outputs, and R_{PI} is the mapping set of an input subset. $v_0 \subseteq V$, where v_0 is the initial state of the control program.

$R_{PI} \subset \beta(PI) \in \{r_i\}$, $(i = 1,2,\ldots)$, where β is the transform function set of confirmed generalized verification. β filters an input subset that is effective in handling state transform. r_i is a subset of PI. PM describes the incomplete mapping of the control program.

$PM(a, b)$: $\beta(V) \times \beta(PO) \times R_{PI} \rightarrow \beta(V) \times \beta(PO)$, where $a \in \beta(V)\times\beta(PO)$, $b \in R_{PI}$. $v_0 \in \beta(V)\times\beta(PO)$.

There are n states inside a controller. Each state transform is decided by M. M is constructed by $\{M_1, M_2,\ldots M_L\}$ inner states. It includes multiple states transferring labeled as φ_i $(i=1,2,\ldots n)$. Each M_1 may have several φ_i. Likewise, the state transformations combined multiple controllers is the same as individual controller inner states. The controller is abstracted by its operation states from the configuration of a controller, e.g. it is showed in Figure 4. So the trusted elements of the controller include input/output interface, configuration transformation, state transferring and programmable components. The network interface is outside the scope of the controller. We embed trusted verification computation in the controller to verify each input/output and state transfer.

The model M of the example consists of four submodule M_i $(i=1,2,3,4)$. $\varphi=\{\varphi_1, \varphi_2, \varphi_3, \varphi_4\}$ is the verified property, which φ_i $(i=1,2,\ldots 4)$ is the state transferring property from the input set to output set. It needs to ensure that the transferring property is trusted from input to output. A group of submodule M_i transforms configuration according to transferring path. We utilize the principle of digital signatures based on Schnorr protocol.[14] The module members are marked particular stamps for output and verified stamps for input.

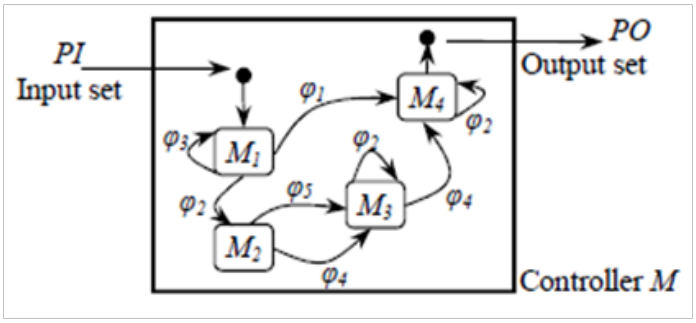

Figure 4 State transferring diagram of controller.

The trusted stamp and the verification protocol

$\varphi_1, \varphi_2, \ldots\varphi_n$ are defined as n state transferring to a controller.

Generates a key pair

To choose two large prime p and q, q is the prime factor of $p-1$. According to Schnorr's suggestion, it is chosen $q \geq 2^{140}$ and $p \geq 2^{512}$, and then the primitive a $(a \neq 1)$ of rank q is chosen in $GF(p)$ and satisfies $a^q \equiv 1 \bmod p$.[14] The one-way hash function HASH is chosen and makes function $h \rightarrow \{0,1,\ldots 2^t\text{-}1\}$. These numbers and the function are promulgated and shared by all state members M_j $(j=1,2\ldots L)$.

Each φ_i chooses the private key K_i which is defined as a random number less than q on $GF(q)$. The public key is

$$U_i = a^{-Ki} \bmod p \ (i=1,2,\ldots n) \qquad (1)$$

Generates state stamp

Assumed that T is the scan period of a response in the controller, state transferring has the performance time T_i

$(i=1,2,\ldots n)$, where $T=T_1+T_2+ \ldots +T_n$. The designed scan period should be abided, or else the design and T_i reset. T and T_i are the time mark of the system given to every member states. It requires that the submodule should finish their work and state transferring at the given times. When state stamp being calculated, the system automatically gets corresponding system unify-time for consistency. Every time T_i exists the differences in precise time because of different performances. The process of state stamp is designed as below that n states transferring to result f.

First, defined $x_0 = a^T \pmod p$. φ_i has choose the random number d_i on $GF(q)$. Calculating

$$x_i = x_{i-1} \cdot a^{-Ti} \cdot a^{di} \pmod p , \qquad (2)$$

and x_i is sent to the sub module M_j $(j=1,2,\ldots L)$ related φ_i. When $j=L$, M_L send x_n to PO.

Then assuming $y_0=0$ and calculating is processed

$$e=h(x_n, f),$$

$$y_i = y_{i-1}+(d_i+K_{ie}) \pmod q \ (i=1,2,\ldots n) , \qquad (3)$$

Citation: Xiao LT, Li MY, Hou KW, et al. A trust verification strategy for autonomous control system in launch site. *Aeron Aero Open Access J.* 2019;3(3):127–132. DOI: 10.15406/aaoaj.2019.03.00090

Finally, $\{f, e, y_n\}$ is sent to *PO* and verified. $\{f, e, y_n\}$ is the group state stamp of the result *f* signed by the state

transferring $\varphi_1, \varphi_2, \ldots, \varphi_n$ in the controller.

Verifying state stamp

The *PO* receives the group state stamp $\{f, e, y_n\}$, and then verifies the whole stamp with the public key U_i of the state transferring. Calculating

$$x_n' = a^{y_n}\left(\prod_{i=1}^{n} U_i\right)^e (\mathrm{mod}\, p) \tag{4}$$

then further verifying $e=h(x_n', f)$ whether or not comes into existence. If the verification equation comes into existence, the output is confirmed that the result is valid and can process the next performance. If the verification fails, the control system refuses the result.

The equation showed below is proved the validity of the state stamp scheme. Because it is based on (3) and (1), it has

$$x_n' = a^{y_n}\left(\prod_{i=1}^{n} U_i\right)^e (\mathrm{mod}\, p) = a^{y_{n-1}+(d_n+k_n e)(\mathrm{mod}\, q)} \cdot \left(\prod_{i=1}^{n-1} U_i\right)^e \cdot (U_n)^e (\mathrm{mod}\, p)$$

$$= a^{y_{n-1}} \cdot a^{(d_n+k_n e)(\mathrm{mod}\, q)} \cdot \left(\prod_{i=1}^{n-1} U_i\right)^e \cdot \left(a^{-k_n}\right)^e (\mathrm{mod}\, p) = a^{y_{n-1}} \cdot \left(\prod_{i=1}^{n-1} U_i\right)^e \cdot \left(a^{r_n}\right)^e (\mathrm{mod}\, p)$$

It can be obtained the below equation on the analogy of (3).

$$x_n' = a^{d_n} \cdot a^{d_{n-1}} \ldots . a^{d_1} (\mathrm{mod}\, p)$$

Due to (2) and initial condition x_0, the equation is

$$x_n' = \left[x_n \Big/ \left(x_{n-1} a^{-Tn}\right)\right]\left[x_{n-1} \Big/ \left(x_{n-2} a^{-Tn-1}\right)\right] \ldots . \left[x_1 \Big/ \left(x_0 a^{-T1}\right)\right] (\mathrm{mod}\, p)$$

$$= x_n \Big/ \left(x_0 a^{-Tn-Tn-1-\ldots . -T1}\right) (\mathrm{mod}\, p) = x_n (\mathrm{mod}\, p)$$

So $e= h(x_n, f)= h(x_n', f)$ is verified.

During the state stamp, every state transferring φ_i apply the below equation to verify the validity of the stamp y_{i-1}.

$$x_{i-1}' = a^{T-T_1-T_2 \ldots . T_{i-1}} \cdot a^{y_{i-1}} \left(\prod_{j=1}^{i-1} U_j\right)^e (\mathrm{mod}\, p)(i = 2,3,\ldots n) \tag{5}$$

The validation is to judge whether or not $x_{i-1}' = x_{i-1}$ comes into existence. In fact (4) is a special example of (5). Similarly, the validation can be capable of proof.

$$x_{i-1}' = a^{T-T_1-T_2 \ldots . T_{i-1}} \cdot a^{y_{i-1}} \left(\prod_{j=1}^{i-1} U_j\right)^e (\mathrm{mod}\, p) = a^{T-T_1-T_2 \ldots . T_{i-1}} \cdot a^{r_{i-1}} \cdot a^{r_{i-2}} \ldots . \cdot a^{r_1} = x_{i-1}$$

During the state stamp, the verification is considered that the group state stamp mentioned above is correct.

Analyzing the security of the trusted stamp protocol

Because a majority of calculation needed to produce a stamp can be finished in the phase of pretreatment and be

independent of massages that are ready to be stamped, the calculation can be processed in idles and not influence the stamping speed. For the same level of security, the stamp length based on Schnorr protocol is much shorter than other signature protocol.[14] The security of the signature and authentication scheme is founded on the computational difficulty of the discrete logarithm. There is randomness influence on trusted operation caused by environment or components performance. Because of state stamping, it is easy to verify these trusted problems during system operation. It will not cause a particular trusted problem. However, intentional attack on the controller is the biggest risk of the trusted operation. The following analysis of the protocol is faced with trusted operation security under attack.

The security of exterior independence attack

The security of every single stamp is the same as the Schnorr's.[15] The attack on group stamp has the modes mentioned below. Under the mode a), b), c), d) and e) the computational difficulty is the equivalence of calculating discrete logarithm on *GF*(*p*).

a. Anyone can obtain the value *p*, *a* and U_i. Someone tries to obtain K_i from $U_i=a^{-Ki}$ mod *p*. It is impossible to obtain the private key K_i that the attacker is directly started with generating the key.

b. Someone tries to obtain K_i or random number d_i by U_i, x_i, and y_i.

c. Someone tries to obtain K_i or random number d_i by the public information $\{f, e, y_n\}$ or $\{f, e, y\}$ and the verification equation (5).

d. An attacker may try to choose randomly an integer K_i' and then solve d_i by the equation of calculating y_i.

e. The attacker has known $x_0, x_1, \ldots x_i$ and tries to solve y_i or has also known y_i to solve d_i which are met the equation

$$x_i = a^{T-T_1-T_2 \ldots . T_{i-1}} \cdot a^{y_i} \left(\prod_{j=1}^{i} (U_j)^e (\mathrm{mod}\, p)\right),$$

who wants to make the whole group stamp met the verification equation by use of calculating a part of group stamps.

An attacker may try to personate or change φ_i under the condition of unknowing K_i. The method is that the history information of the stamping process is collected and then the current stamp information is calculated or personated by the history information. Because the time mark T_i is added to the scheme and different every time, it is different from history and current information. It means that x_i of the same φ_i is different every time and *e* is different too. The history stamp y_i cannot be utilized.

Analyzing the security of united attack

The united attack is that the obtained some stamps are united to solve other stamps of module group. It is not repeated here to the analysis of security similar exterior independence attack. If a part of stamps is united to forge one or several stamps, they need one or several K_i. The condition is impossible as above. If *k* stamps are united to forge *h* stamps of φ_i ($k+h \le n$) and make the group stamp valid to the verification during the state stamping, the verification equation needs to be met. They also face the computational problem of the discrete

Citation: Xiao LT, Li MY, Hou KW, et al. A trust verification strategy for autonomous control system in launch site. *Aeron Aero Open Access J.* 2019;3(3):127–132. DOI: 10.15406/aaoaj.2019.03.00090

logarithm. The replace attack is a kind of forging attack that has much dangerous. Because the security of φ_i's stamp is the equivalence of Schnorr's protocol, the replace attack is invalidated in the scheme of the paper.

It can be summed up the discussion that it is secure to the group stamp based on the computational difficulty of the discrete logarithm. According to the conclusion proved by the theory of computational complexity, 2^t is the difficulty of breaking the protocol (the probability is 2^{-t}, and Schnorr suggested $t \geq 72$).[14,16] The attacker can calculate for several years to forge a stamp.[14,15,17] But time mark is added into the scheme and enhanced the capability of anti-attack. In the state stamp, every module verifies the previous stamp and then stamping after the verification is correct. So the capability of anti-attack is enhanced further.

Conclusion

We have built a prototype system to verify the operation of the protocol. A desk computer connects the controller with the interface of input/output and optical network. The control verification system is carried out on a computer, Intel® Core™ i3 M530 Processor, 2.53GHz, 4GB RAM. The system generates a key pair and sent to package in PI. The verification algorithm modules receive the PO packages and perform the verification on the computer. The controller is chosen ZC-300 PLC based on LOONGSON 1A.[18] The state stamping calculation is embedded in the PLC. The computer sends out input driving at the input port of the controller. Each state transferring of the controller is stamped on the controller. Finally, the computer accepts the output result and the verification module verifies the stamps. For accumulative total about 10000 state transferring, the calculating speed of the stamping and the verification is at most the microsecond order of magnitude or faster. The millisecond-level control is enough to deal with at launch sites. With the controller performance improvement, the application of this protocol has a larger capacity. The research identifies the elements of a trusted autonomous operation based on the control system model in the paper.

The key to the elements is the credibility of the state transferring. We create a guarantee mechanism for trusted operation on the autonomous control system and design the trusted stamp and the verification protocol. By constructing the control system model, the trusted state migration is determined by the model. We design to ensure the trustworthiness of the state migration, so as to obtain the credibility of the autonomous system. The trusted state transferring ensures the autonomous system trusted operation. In the next step, the actual algorithm of state stamp and verification will be optimized. Although theoretical and prototype system is feasible, there are still many issues to be studied in engineering applications.

Acknowledgments

This work was funded by the project of General Technology on Test and Launch partially sponsored by the 973 program. We would like to thank Tsinghua's Prof. Yu Liu who provided us with helpful suggestions and some research resources in the research.

Conflicts of interest

Authors declare that there is no conflict of interest.

References

1. Xiao LT, et al. System Architecture and Construction Approach for Intelligent Space Launch Site. *Advances in Intelligent Systems and Computing*. 2019;856:397–404.
2. Xiao LT, Xiao N, Mengyuan L, et al. *Intelligent Architecture and Hybrid Model of Ground and Launch System for Advanced Launch Site*. IEEE Aerospace; 2019.
3. Xiao LT, Li MY, Ming G, et al. *PLC programs' checking method and strategy based on module state transfer*. IEEE Int Conf Information and Automation; 2015. 702–706 p.
4. Wang J, Shi Y, Peng G, et al. Survey on Key Technology Development and Application in Trusted Computing. *China Communication*. 2016;13(11):70–90.
5. Yudhi Biantoro, Ian Joseph ME. *A literature review of service computing system trust*. 2017 International Conference on Information Technology Systems and Innovation (ICITSI); 2017. 361–366 p.
6. Hiltunen J, Kuusijarvi J. *Trust Metrics Based on a Trusted Network Element*. 2015 IEEE Trustcom/BigDataSE/ISPA; 2015. 19–35 p.
7. Michael JB. Trusted Computing: An Elusive Goal. *Computer*. 2015;48(3):99–101.
8. Zhexiong W, Yu RF, Tang H, et al. *Securing cognitive radio vehicular Ad hoc networks with trusted lightweight cloud computing*. 2016 IEEE Conference on Communications and Network Security; 2016. 1–10 p.
9. Noorman J, Agten P, Daniels W, et al. *Sancus: low-cost trustworthy extensible networked devices with a zero-software trusted computing base*. Usenix Conference on Security; 2013. 479–494 p.
10. Maene P, Gotzfried J, Clercq R, et al. Hardware-Based Trusted Computing Architectures for Isolation and Attestation. *IEEE Transactions on Computers*. 2018;67(3):361–374.
11. Rauter T, Höller A, Iber J, et al. *Thingtegrity: A Scalable Trusted Computing Architecture for Resource Constrained Devices*. International Conference on Embedded Wireless System & Networks; 2016. 23–34 p.
12. Morris T, Vaughn R, Dandass Y, et al. *A retrofit network intrusion detection system for modbus RTU and ASCII industrial control systems*. IEEE 45th Hawaii International Conference on System Sciences: Piscataway, NJ; 2012. 2338–2345 p.
13. Knowles W, Prince D, Hutchison D, et al. A survey of cyber security management in industrial control systems. *International Journal of Critical Infrastructure Protection*. 2015;9:52–80.
14. Schnorr CP. Efficient Signature Generation for Smart Card. *Journal of Cryptology*. 1991;4(3):161–174.
15. Hwang SJ, Hwang MS, Tzeng F. A New Digital Multi-signature Scheme with Distinguished Signing Authorities. *Journal of Information Science & Engineering*. 2010;19(5):881–887.
16. Tahat N, Mustafa Z, Alomari AK. New ID-based digital signature scheme on factoring and discrete logarithms. *Applied Mathematical Sciences*. 2012;6(25-28):1363–1369.
17. Tuan HD, Nguyen HM, Tran CM, et al. *Integrating Multi-signature Scheme into the Group Signature Protocol*. International Conference on Advances in Information & Communication Technology; 2016. 294–301 p.
18. http://www.loongson.cn/product/cpu/1/Loongson1A.html