The Fuzzy Integrated Evaluation of Embedded System Security

Shao Long Zhang , Ning Zhou, and Jia Xin Wu
Center for Studies of Information Resources, Wuhan University, Wuhan， 430072, China
zhangshaolong78@yahoo.com.cn

Abstract

Today, the embedded systems field is growing rapidly, ranging from low-end systems such as cellular phones, PDAs, smart cards to high-end systems such as gateways, firewalls, storage servers, and web server. Security of embedded system becomes a paramount issue in embedded system design. Compared to an embedded system's functionality and other design metric (e.g., area, performance, power), security is currently specified by system architects in a vague and imprecise manner. This paper proposes a fuzzy integrated security evaluation method based on man-computer combined data collection and fuzzy expert evaluation in Delphi method. The method could reduce the subjectivity of expert evaluation and alleviate the difficulty of data collection and makes possible a better combination of qualitative and quantitative evaluations. Firstly, the hierarchy structure model of embedded system security is constructed. Secondly, according to data collected and the evaluation comment of each expert, the subjection degree matrix is constructed. Finally, a new concept of "degree of assurance" is presented for the quantificational evaluation of embedded system security. In this paper a study of a wireless biometric authentication device is also shown. The case illustrates that the method can be easily used and its results conform to the actual situation.

1. Introduction

There is increasing concern of the security threats on these kinds of embedded systems [1] [2], because embedded systems can effect changes in the physical world, the consequences of exploiting their security vulnerabilities can go beyond mere annoyance to significant societal disruption [3]. Successful attacks have been reported on the US Power Grid [4] and the sewer system of Australia's Maroochy Shire Council [5]. Other incidents such as a worm infection [6] have affected the Davis-Besse Nuclear Power Plant and CSX Railroad Corp. To deal with such security threats we need to evaluate the security of embedded system. The security evaluation of embedded system is a process in which the risk factors of the system are analyzed and explained. The basic goal of the security evaluation is to control the risk in acceptability.

Security factors of embedded system include the areas of cryptography, computing, networking and others. However, security is not considered as the addition of features such as specific cryptographic algorithms and security protocols. In reality, the security architecture of embedded system should be more complex. By the reason of the uncertainty of the evaluation factors, the fuzzy logic method is used [7]. In this paper an improved fuzzy method is proposed to solve the security evaluation. The rest of this paper is organized as follows. In section 2, we analyze the security requirements of embedded systems. In section 3, a security evaluation model is given for embedded systems. In section 4, a case is put forward to illustrate our method. Finally, conclusions are drawn in Section 5.

2. Requirements of embedded systems security

Security of embedded systems is different from enterprise and desktop computing because embedded systems perform under constrained resources such as power and memory and are easily accessible at the physical layer.

A common security requirements model of embedded systems is provided [8]. The model considers security from a function centric perspective into system architecture.

978-0-7695-3287-5/08 $25.00 © 2008 IEEE
DOI 10.1109/ICESS.2008.49

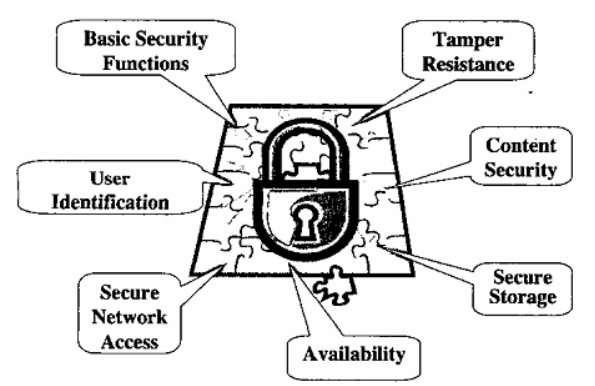

Figure 1. **Common security requirements of embedded systems**

Access to the embedded system should be restricted to a selected set of authorized users (user identification), while access to a network or a service has to be provided only if the device is authorized (secure network access).

The availability of the embedded system from malicious entities could result in a degradation of performance or complete denial of service to legitimate users.

Embedded system security need protect critical or sensitive information (code or data) throughout its lifetime. Secure storage involves securing information in the embedded system's storage devices, external or internal to the system.

Finally, tamper resistance refers to the desire to maintain these security requirements even when the device falls into the hands of malicious parties and can be physically or logically probed.

3. Model of embedded system security evaluation

In the evaluation of embedded system security, we encounter problems such as the complication of security factors and difficulty of data collection. There is no single method to solve these problems. We need an integrated evaluation method which comprises expert evaluation, statistical information and compute technology. To construct security evaluation model, firstly evaluation factors are recognized, then data is collected through man-computer method, at last based on fuzzy subjection theory and Delphi method qualitative evaluation is transferred to quantitative evaluation.

3.1 The recognition of security evaluation factors

Due to these unique characteristics of embedded system security, we can't solve embedded security at a single level of abstraction. With the reference of security pyramid model [9], we give the recognition of security evaluation factors in a three level hierarchy model as following figure.

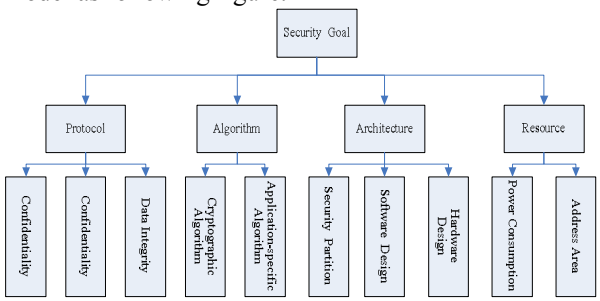

Figure 2. **Three level hierarchy security model**

3.2 Man-computer combined data collection

Reasonable evaluation should be based on data collected. Applied with IT technology including computer log system, automatic audit system, much of security data could be automatically collected such as power consumption and area occupation. Using computer technology can bring benefits. Firstly it could reduce the subjectivity of expert evaluation and make the evaluation method more scientific and objective. Secondly it alleviates the human workload in data collection which not only cuts down the cost of human labor but also improves the veracity and the coverage of data collection.

On the occasions that data could not be automatically collected by computer such as situations about cryptographic algorithm and hardware design, questionnaire survey and On-the-spot investigation also could be applied.

3.3 Expert evaluation in Delphi method

The objective of most Delphi applications is the reliable and creative exploration of ideas or the production of suitable information for decision making. The Delphi Method is based on a structured process for collecting and distilling knowledge from a group of experts by means of a series of questionnaires interspersed with controlled opinion feedback (Adler and Ziglio, 1996). According to Helmer (1977) Delphi represents a useful communication device among a group of experts and thus facilitates the formation of a group judgment [10].

Based on data colleted experts make evaluations relying on their individual competence and are subjective, Delphi method is utilized to adjust the

fuzzy evaluation of each expert to achieve the consensus condition of the all experts consistent.

Flowchart for the Delphi Method follows as [11]:

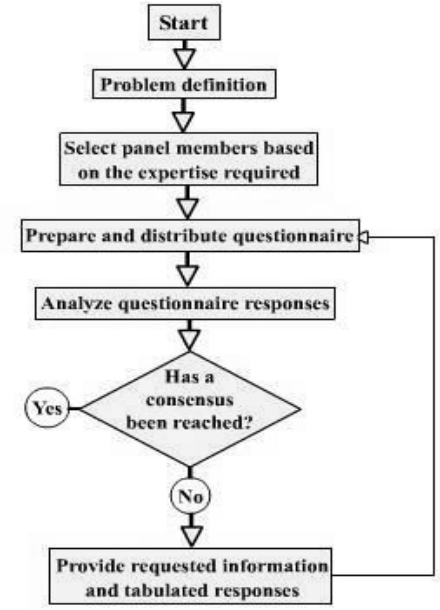

Figure 3. Flowchart of Delphi Method

3.4 Integrated Fuzzy Evaluation

The major steps of evaluation are as the following.

Step1. Construct the hierarchical structure. The top layer is the focus of the goal, and the bottom level, consists of the alternatives under evaluation. The factors and any sub-factors used to make the decision comprise the middle levels [12]. The factors is divided into s subsets which defined as $Y_1,Y_2 \ldots Y_S$ ($Y_i \cup Y_j=\phi(1\leq i,j\leq s\ i\neq j)$). Each subset Y_i is constructed by the factors in the next level denoted as X_{in}, so the characteristic vector of each subset Y_i is presented by the expression.

$$Y_i=(X_{i1},X_{i2},\ldots,X_{in})$$

Step2. Construct the judge set. The expert group which is composed of embedded system experts, embedded system designer and end users provides all the judgments of the set. Suppose the judge set has m judgments, judge set V is $V=\{V_1, V_2,\ldots, V_m\}$.

Step3. Build fuzzy evaluation matrix. We can construct the fuzzy reflection f: $Y\rightarrow F(V)$, Y is the whole of the factor set and F(V) is the whole of the fuzzy set in V. The reflection f means the degree of the support from the factor Y_i to each judgment in the judge set [13]. The subjection vector of Y_i to the judge set V is $R_i=(r_{ij,k})$ n*m(i=1,2,…,s; j=1,2,…,n; k=1,2,…,m). $r_{ij,k}$ is the subjection degree of factor x_{ij} to judgment V_k given by expert which meets the requirements that the range of the value is [0, 1] and the count of values $\sum r_{ij}=1$.

Step4. Estimate the normalized priority weights. The priority weight vector of subsets is presented by the expression.

$$A_i=(a_{i1},a_{i2},\ldots,a_{in}),\text{and}\sum_{j=1}^{n} a_{ij}=1$$

There are several methods such as AHP method and dual correlation function method to give priority weights. According to the particularity of embedded system security, in the case of this paper the weights are given by experts in Delphi method.

Step5. Calculate evaluation vector B_i corresponding to subset Y_i. The calculation formula is

$$B_i=A_i \bullet R_i=(b_{i1},b_{i2},\ldots,b_{im}).$$

Weighted average means is applied to each vector B_i to take valuable information of each evaluation into account. The expression is

$$b_{ik}=\sum_{j=1}^{n} a_{ij}\, r_{ij,k}(k=1,2,\ldots,m).$$

Step6. Calculate the whole fuzzy evaluation. Each subset Y_i is treated as a single element and B_i is treated as evaluation vector of Y_i, The fuzzy evaluation matrix is

$$B=\begin{pmatrix} B_1 \\ B_2 \\ \ldots \\ B_S \end{pmatrix}=\begin{pmatrix} b_{11} & b_{12} & \ldots & b_{1m} \\ b_{21} & b_{22} & \ldots & b_{2m} \\ & & \ldots & \\ b_{s1} & b_{s2} & \ldots & b_{sm} \end{pmatrix}$$

Referred with priority weight vector of each subset A which is represented as $A=(a_1,a_2,\ldots,a_s)$ and $\sum_{i=1}^{s} a_i=1$, the final evaluation $T=A \bullet B=(t_1,t_2,\ldots,t_m)$ is calculated.

3.5 Quantificational evaluation result of security

It is difficult to give quantificational result to final evaluation. To achieve a quantificational evaluation result, the degree of assurance is introduced which is denoted as G.

To calculate G, judge set V should be quantified first. For example, judgment "Very High" would be evaluated as 100. Quantified judge set $V'=\{V'_1, V'_2,\ldots,V'_m\}$.

$$G=\sum_{i=1}^{m} t_i\, V'_i.$$

4. The case

In this paper, we give a case of a wireless biometric authentication device which is to facilitate secure biometric authentication between a user and a server in applications such as intelligent keys, credit card. A user enters acclaimed identity which is stored in the device into a server. After the server validates the claimed identity, the user impresses a fingerprint on the server's sensor. The server extracts the fingerprint's unique features and matches them with a previously stored template. The server decides to corroborate or deny the user's claimed identity based on a matching threshold.

In this case the hierarchy structure is constructed firstly; evaluation factors are divided into four subsets which are Protocol, Algorithm, Architecture and Resource. Sub-factors and the alternatives under evaluation are shown as Tables 1 (priority weight in the brackets).

The judge set V is V= {V_1, V_2, V_3, V_4, V_5} which shows the risk probability level. Its meaning is " V_1 Very High, V_2 High, V_3 Medium , V_4 Low , V_5 Very Low".

Table 1. Evaluation factors of the case

Top factors	Sub-factors	Alternatives
Protocol (0.15)	Confidentiality (0.3)	Challenge–response symmetric key authentication
	Identification (0.4)	Biometric verification
		Session hash authentication code to protect masquerade attack
	Data integrity (0.3)	Generate hash value of encrypt data
Algorithm (0.3)	Application-specific algorithm (0.4)	Feature-extraction algorithm which extracts the minutiae from the raw image
		Matching algorithm which performs a matching operation between these minutiae and a stored template.
	Cryptographic algorithm (0.6)	Advanced Encryption Standard 128 cipher for the encryption on plaintext P with a key K
		Hashing a variable length data stream D to a fixed 128 byte hash application specific algorithm
Architecture(0.4)	Security Partition (0.4)	Partitioning topology that defines how a device is divided into secure and insecure modules
		Coupling and secure-to-insecure bus structure
		Partition functions
		Secure instruction set
	Software design(0.3)	Complexity problem such as buffer overflows using unsafe program languages
		Software vulnerabilities from slipping in as an unwanted extension
	Hardware design (0.3)	Security of processor
		Memory management unit to manage a secure memory space

		Circuitry design for thwarting power analysis attacks
Resource(0.15)	Power consumption (0.5)	Energy consumption for computational requirements
	Address area (0.5)	Increasing cell area for security technology

The experts make judgments of the subjection to the judge set V. Evaluation matrixes R_1, R_2, R_3 and R_4 are:

$$R_1=\begin{pmatrix} 0.15 & 0.40 & 0.35 & 0.10 & 0.00 \\ 0.35 & 0.35 & 0.20 & 0.10 & 0.00 \\ 0.25 & 0.35 & 0.30 & 0.10 & 0.00 \end{pmatrix}$$

$$R_2=\begin{pmatrix} 0.35 & 0.25 & 0.30 & 0.10 & 0.00 \\ 0.20 & 0.35 & 0.40 & 0.05 & 0.00 \end{pmatrix}$$

$$R_3=\begin{pmatrix} 0.15 & 0.25 & 0.25 & 0.15 & 0.20 \\ 0.45 & 0.30 & 0.25 & 0.00 & 0.00 \\ 0.20 & 0.25 & 0.35 & 0.10 & 0.10 \end{pmatrix}$$

$$R_4=\begin{pmatrix} 0.45 & 0.30 & 0.15 & 0.10 & 0.00 \\ 0.35 & 0.25 & 0.30 & 0.10 & 0.00 \end{pmatrix}$$

Calculate evaluation vector Bi

$B_1=A_1 \bullet R_1=$ (0.260 0.365 0.275 0.100 0.000),

$B_2=A_2 \bullet R_2=$ (0.260 0.310 0.360 0.070 0.000),

$B_3=A_3 \bullet R_3=$ (0.255 0.265 0.280 0.090 0.110),

$B_4=A_4 \bullet R_4=$ (0.400 0.275 0.225 0.100 0.000),

The final evaluation T is also calculated as

$T=A \bullet B=$(0.279 0.295 0.295 0.087 0.044)

Referred with the definition in section 3.5, judge set V is quantified as $V'=\{100，80，60，40，20\}$.

The degree of assurance $G=T \bullet V'=73.56$.

Suppose it defines the value G from 0 to 30 to be very dangerous, from 31 to 60 to be unsafe, from 61 to 70 to be normally safe, from 71 to 85 to be well guarded, from 86 to 100 to be perfect, The security of the embedded device in this case is normally safe but it need improves and reinforced

5. Conclusion

Because embedded systems can effect changes in our physical world, the consequences of exploiting their security vulnerabilities can be significant to the society. We need to give the security evaluation of embedded system. In this paper, an integrated evaluation method of embedded system security is presented. Each risk factor is estimated by the experts and for calculating the quantificational security evaluation of the whole system, the degree of assurance is introduced. The case result in this paper shows that the proposed method is scientific and tally with the actual situation.

Acknowledgements

This research is supported by the AOE Importent Project of Philosophy and Social Science. The Project Number is 05JZD00024.It is also support by NSFC under Grant 70473068.

References

[1] Kocher, P., Lee, R., McGraw, G., and Raghunathan, A. 2004. Security as a new dimension in embedded system design. In Proceedings of the 41st Annual Conference on Design Automation (San Diego, CA, USA, June 07 - 11, 2004). DAC '04. ACM Press, New York, NY, 753-760.

[2] Ravi, S., Raghunathan, A., Kocher, P., and Hattangady, S. 2004. Security in embedded systems: Design challenges. Trans. on Embedded Computing Sys. 3, 3 (Aug. 2004), 461-491.

[3] Koopman, P., "Embedded system security," Computer , vol.37, no.7, pp. 95-97, July 2004

[4] F. Schneider, editor. Trust in .National Academy Press, Washington, DC, 1999. Available at http://www.nap.edu/readingroom/books/trust/

[5] Andrew Hildick-Smith, Security for Critical Infrastructure SCADA systems. August 24 2005. SANS Institute. Available at http://www.sans.org/rr/whitepapers/warfare/1644.php

[6] Weaver, N., Paxson, V., Staniford, S., and Cunningham, R. 2003. A taxonomy of computer worms. In Proceedings of the 2003 ACM Workshop on Rapid Malcode (Washington, DC, USA, October 27 - 27, 2003). WORM '03. ACM Press, New York, NY, 11-18

[7] That JHM, Carr V, A proposal for construction project risk evaluation using fuzzy logic" Construction Management and Economics, no.18, pp.491-500, 2000

[8] Kocher, P.; Lee, R.; McGraw, G.; Raghunathan, A.; Ravi, S., "Security as a new dimension in embedded system

design," Design Automation Conference, 2004. Proceedings. 41st , vol., no., pp. 753-760, 2004

[9] Hwang, D.D.; Schaumont, P.; Tiri, K.; Verbauwhede, I., "Securing embedded systems," *Security & Privacy Magazine, IEEE* , vol.4, no.2, pp. 40-49, March-April 2006

[10] The Delphi Method Definition and Historical Background 2007, available at http://www. iit.edu ~it/delphi.html, 2007-11-20.

[11] The Delphi Method 2007, available at http://www.ryerson.ca/~mjoppe/ResearchProcess 841TheDelphiMethod.htm, 2007-11-21.

[12] Mustafa M A, FAI-Bahar J, Project risk assessment using the analytic hierarchy process, IEEE Transactions on Engineering Management, vol.38, no.1, pp.46-52, 1991.

[13] Zhao Dong mei, Comprehensive Risk Assessment of the Network Security, Computer Science, vol.31, no.7, pp66-69, 2004.