Center for Embedded Systems (CES)
Request for Proposals Template – YEAR 5

DUE: Monday, April 8, 2013, by 5 p.m.

TITLE:	Verifiable Firmware Update Mechanisms for Embedded Systems				
PIs:	Ning Weng Spyros Tragoudas	**EMAIL:**	nweng@siu.edu spyros@engr.siu.edu	**TEL:**	618-453-7645 618-453-7027
DEPT:	Electrical and Computer Engineering	**SCHOOL:**	Southern Illinois University		

ABSTRACT:

Maintaining the integrity of embedded systems (ES) is critical but challenging. One of the key challenges is that firmware in ES is expected to be updated post-deployment and malicious code can be injected to ES during process of update. Therefore, a verifiable firmware update infrastructure is required for authorized entities to verify that the software running on remote untrusted ES has not been tampered. This infrastructure must be able to authenticate legitimate users, generate the attestation of code integrity, and deliver them to the designated authorized entity. This project will evaluate varying design solutions for secure update infrastructure in terms of in terms of resilience against varying threats and infrastructure requirements. The possible solutions include but limited to the following: authority-based (back office server) and ES-based (host-monitoring), and hybrid. Evaluated criteria include observed resistance to well-known attacks, system complexity, hardware cost, power usage, administrative overhead. Analysis on real platform will be included in the evaluation.

PROBLEM:

A secure infrastructure is required for authorized entities to verify if the integrity of remote ES in an untrusted zone has not been tampered during firmware update. This infrastructure must authenticate legitimate users, attest code integrity, and deliver to the designated authorized entity.

RATIONALE:

Post-deployment software update presents attacker an avenue for subversion of embedded system [1]. Furthermore, the trend of more frequent major/minor updates is increasing due to the following reasons: more complicated system necessitating more update to fixing firmware bugs, unforeseen new features and requirements to be deployed and increasing Internet connectivity make remote update practical possible attacks during the update. Authorized entities must be able to verify if programs running on remote untrusted devices have not been tampered with by malicious users. The increased interest in remote update will only further emphasize the need for verifiable update mechanisms.

Techniques in software security and computer system security are not directly applicable to embedded systems due to unique vulnerability and characteristics of embedded systems [6]. Embedded systems are usually highly resource constrained, which have limited processing resource for implementations of public-key cryptography. Also embedded systems are generally deployed in highly dynamic and configurable environment, which requires software should be updated with mission of operation changes. Finally embedded systems are working in the autonomous nature and have directly close

interaction with physical entities, which means the failure of embedded systems can have dire consequence.

APPROACH:

Review ES security requirement such as authentication, integrity and access control. Then understand likely ES operational practice (current and future) such as update frequencies after deployment, and partial (eg patches) or full code replacement (eg full flash). Also, enumerate assumptions such as ES resource constraints, the level of trustworthiness of service tool, and connectivity and bandwidth of ES to some remote authority and complexity/overhead of authority. The solution space and quantification metrics of the basic cost of hardening the system against each identified attack can then be qualified. Additionally, reviewing the motivations and goals of a typical adversary will provide a baseline for understanding and quantifying the likely attack methods used and the relative cost of each attack.

Explore evaluate different possible firmware update mechanisms such as Authority-based [2] and Host-monitoring [4]. Authority-based solutions allow firmware that has been explicitly whitelisted to be loaded onto ES. ES has the ability to verify the signature of incoming changes and compare signatures against a signed ticket from remote authority. This method required ES has access to an asymmetric key infrastructure and write-protected storage. A variation of authority-based solution additionally maintains a symmetric key with authority, which is used to decrypt the actual firmware.

Host-monitoring identifies potential attacks on embedded systems. The proposed research will start from an analysis of HIDS architectures including rule-based and anomaly-based solutions. Rule-based HIDS requires the availability of signature with good detection accuracy however fail to address the zero-day attacks. Anomaly-based can detect zero-day attacks but could have high false positive rate. The proposed research will evaluate existing solutions such as watchdog, secure logger as well as enhanced controller [3]. Possible platform to evaluate host-based monitor systems include Freescale MPC564xs [1].

All mechanisms will be evaluated in terms of resilience against primary threats identified above, and also the infrastructure requirement such as ES (hardware, software), ST and back office server. The expected evaluation will include both theoretical analysis and real platform evaluation. A table will summarize the overall evaluation and a final recommendation.

NOVELTY:

- an end-to-end integrity verification mechanisms
- tradeoff between two authority and host-based solutions

POTENTIAL BENEFITS TO INDUSTRY MEMBERS:

- Improved understanding of the total infrastructure costs and benefits of access control systems that employ trusted platform modules or hardware dongles.

DELIVERABLES:

- Description authority-based and host-based verification mechanisms
- Understanding tradeoff varying mechanisms in terms of infrastructure cost
- Final recommendation

TIMELINE/MILESTONES: (PER QUARTER)

Q1 – Review adversary attack methods and verification mechanism specification
Q2 – Authority-based solution
Q3 – Host-based solution, acquire candidate systems for platform evaluation
Q4 – Prepare final report

TECHNOLOGY TRANSFER:

- Biweekly meetings with the sponsor to report ongoing progress and results

BUDGET:

$35,000 is requested to support the PIs and graduate students, to purchase demonstration hardware, and trips to the industrial liaison site.

BIBLIOGRAPHY:

[1] Ang Cui, Michael Costello, Salvatore J. Stolfo; "When Firmware Modifications Attack: A Case Study of Embedded Exploitation;" *NDSS 2013*; 2013

[2] Basile, C.; Carlo, S.D.; Scionti, A., "FPGA-Based Remote-Code Integrity Verification of Programs in Distributed Embedded Systems," *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on* , vol.42, no.2, pp.187--200, March 2012

[3] SHE: Secure Hardware Extension, https://www.escrypt.com/fileadmin/escrypt/pdf/WEB_Secure_Hardware_Extension_Wiewesiek.pdf. . [Accessed 12 April 2013].

[4 Mao, S.; Wolf, T., "Hardware Support for Secure Processing in Embedded System," IEEE Trans. Computers 59(6): 847-854 (2010)

[5] Freescale, " MPC564xS: Qorivva MCU for instruments cluster," 2013. [Online]. Available: http://www.freescale.com/webapp/sps/site/prod_summary.jsp?code=MPC564xS. [Accessed 12 April 2013].

[6] S. Ravi, A. Raghunathan, P. Kocher, and S. Hattangady. Security in embedded systems: Design challenges. ACM Trans. Embed. Comput. Syst., 3(3):461–491, 2004.

I/UCRC Executive Summary - Project Synopsis		Date: 2013-4-5
Project Title: **Verifiable Firmware Update Mechanisms for Embedded Systems**		
Center/Site: SIUC		
Principle Investigator: Ning Weng and Spyros Tragoudas		**Type: (New or Continuing) New**
Tracking No.: (CES office to input)	**Phone :** 618-453-7645;618-453-7027	**E-mail :** nweng@siu.edu;spyros@engr.siu.edu
		Proposed Budget: $35000

Abstract: Maintaining the integrity of embedded systems (ES) is critical but challenging. One of the key challenges is that firmware in ES is expected to be updated post-deployment and malicious code can be injected to ES during process of update. Project will evaluate varying design solutions for secure update infrastructure in terms of in terms of resilience against varying threats and infrastructure requirements. The possible solutions include, but not limited to, the following: authority-based (back office server) and ES-based (host-monitoring), and hybrid. Evaluated criteria include observed resistance to well-known attacks, system complexity, hardware cost, power usage, administrative overhead. Analysis on real platform will be included in the evaluation.

Problem: A secure infrastructure is required for authorized entities to verify if the integrity of remote ES in an untrusted zone has not been tampered during firmware update. This infrastructure must authenticate legitimate users, attest code integrity, and deliver to the designated authorized entity.

Rationale / Approach: Post-deployment software update presents attacker an avenue for subversion of embedded system. Frequent major/minor updates trending up; authorized entities must be able to verify if programs running on remote untrusted devices have not been tampered with by malicious users.
Explore evaluate different possible firmware update mechanisms such as Authority-based and Host-monitoring.

Novelty: An end-to-end integrity verification mechanisms; tradeoff between two authority and host-based solutions.

Potential Member Company Benefits: Improved understanding of the total infrastructure costs and benefits of access control systems that employ trusted platform modules or hardware dongles.

Deliverables for the proposed:

- Description authority-based and host-based verification mechanisms
- Understanding tradeoff varying mechanisms in terms of infrastructure cost
- Final recommendation

Milestones for the proposed year:
Q1 – Review adversary attack methods and verification mechanism specification
Q2 – Authority-based solution
Q3 – Host-based solution, acquire candidate systems for platform evaluation
Q4 – Prepare final report

Progress to Date: **THIS SECTION TO BE UPDATED IN JANUARY**

Estimated Start Date: 8/15/2013	**Estimated Knowledge Transfer Date**: 8/31/2014