Security Concepts and Issues in Intra-Inter Vehicle Communication Network

Mustafa Saed, Scott Bone, John Robb
Hyundai-Kia America Technical Center, Inc.
Automotive Company
Superior Township, MI 48198, USA
{msaed, sbone, jrobb}@hatci.com

Abstract—**It is demanding to provide secure communication among vehicles in Vehicle to Vehicle (V2V) and Vehicle to Infrastructure networks (V2I). Vehicles need to authenticate each other and verify the integrity of the shared safety information which is critical. Adversaries can masquerade as real subscribers in V2V/V2I networks and broadcast bogus messages before to destroy the system with such as sending inaccurate safety information to other vehicles. The intent of this paper is to survey the attempts that have been made to tackle vehicle security, and present the security approaches necessary to enforce tough security measures that fully protect the vehicle security architecture.**

Keywords—**V2V, V2I, CAN Bus, Security Network, Security Threats**

I. INTRODUCTION

In the past decade, automotive companies that develop telematics systems have been faced with critical security and privacy issues related to everyday applications that allow interfacing between vehicles and humans. Interaction between people and vehicles will lead engineers to creative thinking of ideas and quite possibly a paradigm shift in determining the methods to provide a sufficiently secure system that will cover all the accessible gaps to unauthorized users. . Security concerns were less prevalent in the past due to technological gaps, but with advancements in technology (e.g. ability to develop ad-hoc interfaces, easily accessible hardware and software, etc…) every day computer hackers and cryptologists have created a tremendous amount of concern. Current vehicle architectures are at risk for wireless security break-ins, but future vehicle architectures and systems will increase the risk and this risk needs to be mitigated. These risks will be enabled with the use of embedded phones and wireless protocols containing private information (e.g. financial records, pin numbers, credit card information, birth dates, etc…) [1]. Protecting the customer's private information and the vehicle systems from "hackers" and the infectious viruses these software programs produce will have a direct effect on trustworthiness and the quality from the view of the consumer as well as the vehicle's safety dynamics, such as the multiple Electronic Control Units (ECU's) and its associated systems relying on accurate and uncorrupted information.

The most probable scenario for hacking to occur is to take full advantage of the telematics /wireless feature embedded in the vehicle and performing the function of the electrical system brain; therefore, allowing this module to become the "open input" to the world. There are currently 2 identifiable probable solutions in closing the technological gap:

- Inter- Vehicle Communication: Secure the communication/protocol between the vehicle and the infrastructure through the wireless network.
- Intra- Vehicle Communication: Secure the communication/protocol between the telematics unit and the ECUs connected through the CAN Bus.

The first solution Inter-Vehicle communication incorporates the use of cryptography and data security with the packet data session over (TCP/IP) and the voice service. The planned proposal(s) include piggy backing off the developed concepts that have been somewhat successful in the security arena. There are a number of researches that are attempting to secure the V2V Networks, but these are still not strong enough to provide effective security and safety. There have been many attempts regarding this technique, such as:

Roshan Duraisamy et al [2] proposed a new hardware implementation, which uses Elliptic Curve cryptography and Digital Signature Algorithms (ECDSA), by enabling two parties "a remote agent and network embedded system" to initiate a 128-bit symmetric key, and make all transmitted data encrypted via the Advance Encryption Scheme (AES). Chenxi Zhang et al [3] introduced a new technique Identity-based Batch-Verification (IBV), which uses a private key for pseudo identities; therefore, the certificates are not required. Each received signature will be verified within 300 ms intervals, but this depends on the Dynamic Short Range Communication (DSRC) protocols. Yi Qian et al [4] proposed how much Medium Access Control (MAC) the layer protocol can achieve through both Quality of Service (QoS) and security requirement for vehicular networks safety application, and designing of efficient MAC protocol to achieve the safety related vehicular networks.

Based on the techniques introduced above, there is a critical concern with securing the vehicle to infrastructure communication. This is due to the fact that all communication between the vehicle and roadside units are implemented with wireless technology; thus, allowing for the probability of numerous attacks or viruses being injected into the unprotected system if security is not enforced. Avoiding these problems and creating a secure,

effective, and yet useful Inter-Vehicle communication method will ensure system reliability [5].

The second solution, Intra-Vehicle Communication requires the protection of data transmission between the vehicle ECU's through the Controller Area Network (CAN) Bus which is an open and unsecured automotive protocol. To this point, automotive companies haven't had concerns with securing this type of communication due to the low risk of infiltrating the CAN Bus remotely. In fact, the only way of accessing the CAN Bus is by connecting a diagnostic tool physically to the vehicle through an On-Board Diagnostic (OBD) connector, so that authorized technicians can perform troubleshooting analyses. However, with the ability to easily develop hardware interfaces and software application layers, automotive companies are implementing to access the CAN Bus remotely through the telematics ECU by using Wi-Fi, BT and cellular network. With this possibility existing, security risks are now increased to the point of allowing "unauthorized systems and network access, Auditability and compliance, Customer data breaches, Internal and external sabotage, and the Theft of intellectual property and confidential business information" [6]. This paper will present background into the specifics of CAN applications, provide the work that has been done so far in the field of vehicle security, and then the security recommendations in the vehicle security field. The rest of the paper is organized as follows: Section 2 focuses on automotive multiplexing methods, classifications and protocols. Possible vulnerabilities in vehicle communication are presented in section 3. Section 4 discusses vehicle communication security issues. The future of vehicle security is introduced in section 5. Finally section 6 concludes the paper.

II. Automotive Multiplexing Methods, Classifications and Protocols

A. Multiplexing Methods and Classifications

Multiplexing in automotive technology has become the greatest achievement in the struggle to make vehicles more efficient by reducing the weight in the power distribution system (i.e. wire harnesses) and keeping modules embedded for security. Multiplexing in terms of technological specifics utilizes a single or dual wire (i.e. bus) connecting multiple ECU's and their corresponding messages and signals through two primary methods; time division or frequency division process. The time division strategy inserts a sample of each channel onto the data stream and the channels are selected for a short period of time. This uses the most accurate form of time sharing amongst various channels and is the method most prevalent in the automotive industry [7]. Frequency division, however, uses a different approach which shares the process amongst various channels where information can be designated by a carrier frequency via each channel to modulate the sinusoidal signals [7].

For the purpose of accurately determining the protocol for developing multiplexing strategies between the various vehicle sub-systems, the Society of Automotive Engineers (SAE) divided the automotive communication sector into three classes. These classes are described as:

- Class A can support 100 nodes and is categorized to handle data speeds (i.e. baud rate) up of 1 kilobit per second (kb/s). However, the lag time, which is the time delta between a transmission request and transmission initiation, is 50 ms. Class A baud rate is used in the following systems: tail light, turn signals, driver convenience features, and entertainment systems [7].
- Class B can support 50 nodes as it is categorized as an information system with data speeds up to 100 kb/s [7].
- Class C is mainly used in real time events that require urgent speed with high accuracy values. Its data rate is in up to 1 Mb/s. Class C baud rate is used mainly in powertrain systems. Class C does not accommodate new systems such as Intelligent Vehicle Highway System (IVHS), collision avoidance system, Global Position System (GPS), and many other related systems [7].

The various types of communication signals are transmitted and received by many types of network nodes; known as protocols. These protocols are created by a set of rules for coding, address structure, transmission sequence, error detection, and handling. When associated with automotive networking, protocols cover a majority of functions assigned to the various layers of the Open System Interconnection (OSI) model. When involved in a noisy surrounding, a multiplexing protocol would be optimized to meet the technical and functional specifications of the system.

B. Protocols

Inter-Controller Area Network (ICAN) is a network protocol designed primarily for the vehicle networking environment. A CAN controller acts as mediator to alleviate the node processor from over-working itself from the high speed of message transfer. In CAN, disputes between messages are determined on a bit-by-bit basis in a non-destructive arbitration, which result in the highest priority message gaining access to the bus. The CAN protocol supports 2,032 different messages of up to 8 bytes of data. Unlike many serial communication protocols, CAN message data contains no information related to the destination address. The message contains an identifier which indicates the type of information contained within. This feature allows for convenient addition or deletion of the intelligent nodes in an automotive system. Also, each node decides whether to read or ignore a CAN message. A message may broadcasted to multiple nodes by using the CAN protocol [7].

III. VULNERABILITIES IN VEHICLE COMMUNICATION

Most products are designed to stop good people from unintentionally doing bad things. This has led to situations in which product security is sometimes an afterthought resulting in frequent redesigns. Making security decisions as early as possible in the design phase makes it easier to avoid costly redesigns that are difficult to both manage and implement.

Threat Modeling, an essential design practice used during all stages of product development, is a practice that can be used to ensure that all security threats have been realized, documented, and mitigated. This practice can also help device makers ensure all stakeholders have considered security as part of the overall product design and part of the developmental process. Since products are often built by several parties, successful use of Threat Modeling requires that all involved parties adopt this practice [8].

Threat Modeling usage scenarios define the scope of the design. All possible usage scenarios including any scenarios perceived as out-of-scope should be listed and marked accordingly. Scenarios should cover all features used by the system, not just the scenarios used by a car. The following are examples of usage scenarios:

- A car used by a home user
- A car used as a taxi
- A car used as a rental device
- Car Wi-Fi connected to a home access point
- Car Wi-Fi connected to a home access point and roams onto public hotspots
- A car at a dealership used as a demonstration vehicle
- User installs third party electronics device on a CAN bus

A definitive list of Usage Scenarios allows all development process stakeholders to know how the device can and will be used, and this also helps teams identify scenarios not previously considered. The following examples are threat categories found with mitigation strategies defined:

- Threat the categories: Tampering, information disclosure, and denial-of-service, elevation of privilege: Address book entries are sourced from untrusted external sources and stored in a user's address database. External sources include; USB devices, Memory cards, Bluetooth technology, Wi-Fi, HMI editing, Internet (navigation traffic data), and eCards.
- Threat categories: Information Disclosure: Device crash dumps and device logs are memory and file system dumps whose primary purpose is to aid system debugging, which may contain Personally Identifiable Information (PII) such as phone numbers and SMS text messages. Such files may be obtained from customer vehicles in the field to debug difficult to replicate or high severity issues. If PII is included in those files, it can be viewed by parties outside of the private individual the crash dumps were taken from. Crash dumps, including PII could be supplied to 3rd party application developers by the Vehicle Manufacturer for review, possibly unintentionally disclosing PII [9].

IV. SECURITY ISSUES AND THREATS IN VEHICLE COMMUNICATION

While the need for advanced telematics systems continues to drive consumer interest, automotive manufacturers are equally pressured to provide workable systems that can guarantee no unauthorized entry from hackers and other cryptology experts. These types of concerns were never a problem in the past due to gaps in electrical knowledge and technology, but with the sudden advancements in IT, and the ability to develop ad-hoc interfaces with easily accessible hardware and software, developers are suddenly overwhelmed with concern fueled by the increased knowledge and capabilities of the average hacker. These concerns are focused on some of today's vehicles, but even more so in regards to the development of future vehicles.

A. Boot Loader

At the time of power-on of any embedded system a Boot loader will be invoked directly from memory. The security issues for the boot loader update process can be addressed as follows:

- An update process that can be used to downgrade as well as upgrade any components (i.e., install v1.2 code over already installed v1.3 component) could be used by a user to install a less secure component, making the device easier to exploit
- The update process can be used to update the Boot loader itself.
- The update requires settings to be changed or recalculated (updating certificates, or updating stored security hashes).
- If the update fails part way through the update process (due to loss of power, failed write to memory or Denial-of-Service) the install process requires a means to back out any changes.
- If any part of the update fails, the update will leave the device vulnerable or malfunctioning. An example of this situation is an update that contains security certificates that need to be stored in a hardware security store (a special area of memory, or separate memory entirely that is not accessible via the data/address bus, such that applications are not able to read/write to it) and files that are signed with the new certificate.
- The update process keeps a log of features updated and configuration changes. The update process modifies user data.

B. Privacy

Personally Identifiable Information (PII) is any information such as one's name and phone number that can be used to distinguish or trace an individual's identity. Information that can be linked to an individual such as location, favorite shops, and music is also classified as PII. For devices utilizing PII, the recommended security approach is to first consider all information private and be able to be associated with a person or individual object and to then justify why any data is considered not private. PII must always be secured, but note that secured does not necessarily mean disclosed. PII may be disclosed, but only with the individual's knowledge and consent [10].

C. Operating system OS

Most common embedded operating systems are those that have been thoroughly developed and designed by large corporations such as Microsoft® (Microsoft Windows Embedded, Windows Embedded Automotive) and QNX (QNX CAR Application Platform). The security issues for the operating system can be addressed as follows:

- Developers add their own custom encryption or wrap an existing encryption type in their own code leading to unforeseen weaknesses in encryption and security problems.
- Developers add their own versions of standard functions (strcmp(), strcpy(), memcpy(), etc.). Custom implementations may include quirks that can lead to unforeseen performance, stability, and security issues. Developers misuse high risk functions (for example: sscanf(), strcat()). Incorrect use will not be detected via build warnings [11].

D. Application

Application types can include native applications, Java Virtual Machine (JVM) and Adobe Flash Player (FP) applications. These application types differ, in-part, based type of device access. Typically, native applications will be written because they need a higher level of access to the device or because of performance reasons. The security issues for the application program interface (API) can be addressed as follows:

- API functions call directly into hardware devices
- API functions allow access to system configuration files
- API functions trigger other system applications to be executed. This can lead to privilege escalation threats and vulnerabilities
- API functions have access to system events. If events are influenced by the application rather than the API, unforeseen instabilities and threats arise within the application as well as in system services [11].

E. Communication

Any device that connects to external sources whether trusted or untrusted sources will have inherent security threats that need to be mitigated. Many of these threats are not device specific but are effectively communications specific Therefore, whenever reviewing security aspects of communications systems, extreme care should be taken to not make any assumptions about anything. Many communications systems and protocols were designed and developed before secure design techniques existed (these practices were developed because of the lack of security in design and development). Also, adding security at a later date often does not enhance the security of a system. Accordingly, it is easy to understand why communications systems pose the highest security threats to a system. The security issues for the communication can be addressed as follows [11], [12], and [13]:

- Connection to unauthenticated user – protocols such as ARP and DHCP do not authenticate the server to which they are connecting. Connection is usually assigned by the quickest response. On networks, a malicious device, if able to respond quicker than the intended server, can cause the target to connect to it rather than the intended server, this type of threat can lead to information disclosure, Denial-of-Service, spoofing and tampering.
- Name resolution services are easily confused – typical systems use domain name servers (DNS) to identify target machines by name rather than an IP address, these services can easily be exploited and have had multiple vulnerabilities in them.
- Network Bridging – When multiple methods to connect to the Internet are possible (Wi-Fi, cellular, Bluetooth technology, USB modem), it is possible that several connections may be active at the same time. This is an extremely risky practice do to name resolution issues and other vulnerabilities may occur as a result of the simultaneous connections.
- Weak configuration authentication: many of the Internet configuration settings are accessible to the whole system. An application may be required to setup the configuration of another application or service, but the system may require a different configuration. Thus, an application needs to be held responsible for ensuring it has the correct configuration at all times. Often, all applications are given access to facilities to release and/ or renew the device IP and other network configuration items, thus, allowing them to disconnect other services that require specific connections.
- Local host easily exploitable – many applications in the past used the local host IP for inter-process communications (IPC) since unauthenticated user applications may be able to gain access to or block access to these services. They may also be able to gain elevation of privilege or cause a Denial-of-Service to system features.

- Poorly implemented networking code – Code, if badly written to perform mutual authentication over SSL, can leave a system vulnerable to man-in-the-middle attacks.
- Blocking sockets – Poorly written code utilizing blocking sockets can cause local Denial-of-Services threats and vulnerabilities.
- Raw sockets – Raw sockets allow applications to see all network traffic on a device. Some of this information may include security or private data, which can be used to exploit other vulnerabilities. Badly implemented raw sockets could also open up other applications to see additional data causing stability issues and security vulnerabilities.
- Shortened URLs – Extreme care should be taken when developing, reviewing and using these URLs since they are frequently used to direct the user to install malware or to download a virus. Shortened URLs are most commonly used on social networking sites.
- ASCII/Unicode Threats – Many Internet protocols (example http) were developed based on these character sets. Many threats and vulnerabilities exist through exploitation of interpretation of character strings that contain ASCII control codes and non-displayable characters.
- Network Firewall rule errors – Poorly implemented firewall rules or applications being able to modify a network firewall can render the firewall useless and expose the system to further exploits. A common rule error is caused when the rule priority or order of execution is modified to insert an allow-all rule somewhere in the rule sequence.
- Data cost/charge – Many Wi-Fi and cellular data systems have an associated cost of usage, either monthly or per mega-byte. Any external data usage should be controlled by the system such that the user does not incur unexpected charges due to extreme data usage [13].

F. USB

It is (currently) a wired technology that allows many different device types to be connected to the system. Typical USB devices include: cameras and web cameras, flash memory cards, Wi-Fi adapters, Bluetooth technology interface, cellular internet adapters, hubs, phones, media, players, personal computers, mice and keyboards, GPS devices, serial port devices and multifunction adapters. The security issues for using the USB can be addressed as follows [10]:

- Device insert/ejection – USB devices can be inserted or ejected at any time. When any service is utilizing the feature, care should be taken to ensure no disruption of service. A malicious user can repeatedly insert and then remove a USB device at a rapid rate causing the system to go into an unstable state and a Denial-of-Services. Performing this action while the vehicle is in motion could cause driver distraction and an accident; therefore, device detection may be restricted to when a vehicle is stationary. So hacking the system will cause safety issues.
- USB Flash Memory devices –Adding flash memory devices to the system add areas to the file system. Extreme care should be taken since any contents should be considered untrusted with no guarantee of reliability can be assumed. With continued usage device sector read and write failures can occur, when any system application reads or writes to a device, it should protect against this type of failure.
- USB Multifunction devices – A current trend in USB devices is to provide, for example, a USB cellular modem with build in flash memory device (and a micro SD card slot), the flash memory device usually contains auto-runnable code to install a driver for the cellular modem. Since the modem would typically be manufactured for use on a desktop system, it is highly unlikely that the available drivers would operate on the vehicle system. Also, the USB modem memory card could have been modified by a malicious user and replaced with malware or a virus.
- Multiple identical USB devices – It may be possible to connect two USB Flash memory devices to the system, but developers should be aware of security issues arising from device insertion order. It is also not frequently accounted for that two USB communication devices are connected at the same time (e.g., two cellular modems).

G. Wireless

The security issues for the Wireless can be addressed as follows [10]:

- Wireless connection information is stored in weakly protected databases that can be accessible to untrusted applications. If an untrusted application is given full access to the database where all previous wireless networks are stored. Existing entries can be deleted, Existing entries can be tampered with and new entries may be added. This may lead to a Denial-of-Service or granting access to an untrusted network.
- Wireless connection passwords and keys are stored in clear text. If wireless keys are stored in clear text and untrusted applications have the ability to read the data, further exploitation is possible.
- To allow greater areas to be covered by Wi-Fi, it is a common practice for a network to have multiple access points (AP's). Each AP will contain an identical AP name, but have a different MAC address. When the Wi-Fi device moves from an area, where the signal is currently connected to is stronger than the other Wi-Fi AP, to a location where the current connection is weaker, then it will attempt to connect to the stronger signal. Security considerations need to be enforced when this occurs. A number of questions need to be asked: is the new Internet path as secure as the previous one? Have any security protocols changed as a result of the change in AP? Can the new AP be trusted?

Often the transfer from AP to AP occurs internally to the Wi-Fi subsystem and is transparent to a user. However, consideration of this needs to be made during threat modeling and system development. Wireless roaming also exists between different network types. Some examples include, Wi-Fi to Cellular Internet USB modem, Wi-Fi to Bluetooth PAN Internet, Bluetooth PAN Internet to Cellular Internet USB modem, and Wi-Fi home network to a Coffee Shop Public Wi-Fi network. Each wireless network type has individual security implications that need to be considered.

V. THE FUTURE OF VEHICLE SECURITY

The future of vehicle security seems promising. The following security improvements should be taken care of [14], [15], [16], [17], [18] and [10]:

- Enhancing the vehicle security approach by adopting the Internet protocol version 6 (IPV6) in the vehicle communication protocol, synchrophasor security/NASPInet, anonymization, behavioral economics/privacy, and cross-domain security involving it.
- Using the public key infrastructure (PKI) in the vehicle security, and addressing all the related security requirements of the operation and devices of the vehicle communication.
- Securing the trusted device profile and implementing and developing the vehicle security certificate lifetime.
- Resolving the privacy concerns regarding customer information in the vehicle.
- Preventing the transfer of some critical data, such as the business location or cross border data transmission.
- Implementing a robust security approach for the vehicle communication as a future priority to achieve proper authentication in any device communication via the vehicle.
- Addressing all the newly created vulnerabilities of the vehicle communication by monitoring and tracking the communication and the data flow through the vehicle.

VI. CONCLUSIONS

This paper presented the work that has been done so far in the field of vehicle security. It also introduced the future approaches, techniques, and methods needed to improve and enhance this security. All of the security features that the vehicle needs to cover were addressed. The paper provided a broad view of how we can make the vehicle a very secure system to take full advantage of all its features. Our future work will focus on extending security requirements to all the vehicle communication, including in vehicle communication, vehicle to vehicle communication, vehicle to infrastructure communication, and third party access. The main focus will be on how to implement powerful cryptographic protocols to achieve outstanding security.

REFERENCES

[1] S. Lee, G. Pan, J. Park, M. Gerla and S. Lu, “Secure incentives for commercial and dissemination in vehicular networks,” in Proc. the 13 Annual International Conf. Mobile Computing and Networking, 2007, pp. 150-159.

[2] Roshan Duraisamy, Zoran Salcic, Maurizio Adriano, and Miguel Morales-Sandoval, “Supporting Symmetric 128-bit AES in Networked Embedded Systems: An Elliptic Curve Key Establishment Protocol-on-Chip,” University of Auckland, University of Rome, National Institute for Astrophysics, Optics and Electronics, 2006.

[3] Chenxi Zhang, Rongxing Lu, Xiaodong Lin, Pin-Han Ho, and Xuemin (Sherman) Shen, “An Efficient Identity-based Batch Verification Scheme for Vehicular Sensor Networks,” University of Waterloo, 2008.

[4] Yi Qian, Kejie Lu, and Nader Moayeri introduced paper, “Performance Evaluation of Secure MAC protocol For Vehicular Networks,” National Institute of Standards and Technology, University of Puerto Rico, 2008.

[5] U.S Department of Transportation, National Highway Traffic Safety Administration, "Vehicle Safety Communications Project; Task 3 Final Report; Identify Intelligent Vehicle Safety Applications Enabled by DSRC," Notional Technical information service (22161), Virginia, March 2005.

[6] William Stallings, “Cryptography and Network Security Principle and practices,” New Jersey, NJ: Pearson Prentice Hall, 2010.

[7] Paret, D. and Riesco, R., "Multiplexed Networks for Embedded Systems: CAN, LIN, FlexRay, Safe-by-Wire," SAE International, June 20, 2007.

[8] Koscher, K., Czeskis, A., Roesner, F., Patel, S. et al., “Experimental Security Analysis of a Modern Automobile,” IEEE Symposium on Security and Privacy, 2010, pp. 447–462.

[9] Checkoway, S., McCoy, D., Kantor, B., Anderson, D. et al., “Comprehensive experimental analyses of automotive attack surfaces,” Proc. of USENIX Security, 2011.

[10] Forouzan, B., "Cryptography and Network Security," New York, NY: Mc Graw Hill, 2008.

[11] The Telegraph, “Thieves placed bugs and hacked onboard computers of luxury cars,” 02 July 2012. Available: http://www.telegraph.co.uk/news/uknews/crime/9369783/Thievesplaced- bugs-and-hacked-onboard-computers-of-luxury-cars.html

[12] Wright, A., “Hacking cars,” Communications of the ACM, Nov. 2011.

[13] Bouard, A., Schanda, J., Herrscher, D., and Eckert, E., “Automotive proxy-based security architecture for CE device integration,” Proc. of Mobileware, 2012.

[14] Zagar, D. and Grgic, K., “IPv6 security threats and possible solutions,” WAC, July, 2006, pp. 1-7.

[15] Zhao, M., Smith, S., and Nicol, D., “Evaluating the Performance Impact of PKI on BGP Security,” PKI Research and Development Workshop, Gaithersburg, 2005.

[16] Wolf M. and Gendrullis, T., “Design, Implementation, and Evaluation of a Vehicular Hardware Security Module,” ICISC , 2011.

[17] Hersteller Initiative Software, “SHE Secure Hardware Extension V1.1,” 2009. Available: http://www.automotive-his.de

[18] Hoppe, T., Kiltz, S., and Dittmann, J., “Security Threats to Automotive CAN Networks Practical Examples and Selected Short-Term Countermeasures,” Computer Safety, Reliability, and Security, 2008.