Interoperable Medical Instrument Networking and Access System with Security Considerations for Critical Care

Deniz Gurkan*
College of Technology, University of Houston, 4800 Calhoun Rd., 230B Technology Bldg, Houston, TX 77204-4020. dgurkan@uh.edu

Fatima Merchant
College of Technology, University of Houston, 4800 Calhoun Rd., 394 Technology Bldg., Houston, TX 77204-4020. fmerchant@uh.edu

ABSTRACT
The recent influx of electronic medical records in the health care field, coupled with the need of providing continuous care to patients in the critical care environment, has driven the need for interoperability of medical devices. Open standards are needed to support flexible processes and interoperability of medical devices, especially in intensive care units. In this paper, we present an interoperable networking and access architecture based on the CAN protocol. Predictability of the delay of medical data reports is a desirable attribute that can be realized using a tightly-coupled system architecture. Our simulations on network architecture demonstrate that a bounded delay for event reports offers predictability. In addition, we address security issues related to the storage of electronic medical records. We present a set of open source tools and tests to identify the security breaches, and appropriate measures that can be implemented to be compliant with the HIPAA rules.

Keywords: IEEE 11073, medical instrument networking, CAN, HIPAA, security of medical data.

1. INTRODUCTION
Critical care requires complex medical instruments to report and record patient health in a timely and accurate manner. Current technology and health record systems rely on these instruments to be utilized by the healthcare personnel, who then record health information into an electronic system. This mandates that a healthcare personnel be at the bedside of a critical patient full-time, to enable continuous care. Moreover, manual entry of health information into an electronic system by healthcare personnel is prone to error. Therefore, a **bottom-up approach** to realizing health record monitoring, archiving, and access should be developed that utilizes instrument data to be posted on the electronic system without the need of a healthcare professional's manual entry.

*Corresponding author

Automatic transmission of medical data from an instrument is possible with proprietary mechanisms that vendors provide for each instrument. However, the critical care room is typically equipped with instruments from various vendors. Therefore, a uniform access point to all patient data is impossible without interoperability among vendor application interfaces. **This paper proposes an interoperable networking of all critical care bedside instruments, with a secure access mechanism to all patient health records.**

2. PREVIOUS WORK

There has been an emerging interest in combining the strength of the communication networks with monitoring capabilities of medical devices in order to deliver an automated support system for medicine. On one hand, standardization efforts such as the IEEE 11073 led the way towards a unified data structure among the point-of-care devices. On the other hand, there were numerous demonstrations of wireless and wired networked medical devices/instruments. Previous works can be categorized into three major areas as far as this paper's focus is concerned: (i) networking and standardization, where interoperability of individual medical instruments on a network environment has been explored; (ii) isolated automation demonstrations, where integrations of medical instrument and patient care have been proposed towards automated care; and (iii) automation and security of medical health records.

Networking and standardization: One study demonstrated a system with a complete requirements list to amplify the effectiveness rather than swapping of the staff with an automated data delivery protocol [1]. More work has been done on incorporation of radio frequency (RF) identification technology into a complete system delivery for healthcare [2] as well as data acquisition from instruments again using a standard management information base [3,4]. The interoperability issues of these standards have been extensively studied towards creation of a relational database tree among the relevant monitoring parameters [5]. However, when the standards are not followed by industry products because of their complexity and an uncooperative environment, they fail to emerge as an acceptable method. The demonstrations in the research laboratories are promising to the extent of a proof-of-concept [6]. These demonstrations are desired to be picked up by the industry to gain momentum in a successful standardization process. Other open standards have been utilized such as the fieldbus technology to build patient monitoring systems [7]. This work outlined an extensive list of requirements for bedside monitoring and signal representation in terms of ease of use. In order to further the standardization efforts, a study on medical instrument companies and their family of products has been investigated [8,9]. These studies have been focused on the technical details of interfacing specific medical instruments, without attention to standardization of the medical data that are reported. For example, signals captured can be in the form of **monitoring parameters such as the heart rate, events such as an alarm indicating increase in blood pressure, or data streams** such as plethysmograph. The feasibility of telemedicine and standardization efforts with cost-benefit ratios has been studied for European as well as American healthcare needs [10]. The results show an exceptionally high benefit in automation and communication of the medical instruments through a network.

Isolated automation demonstrations: An anesthesia alarm system was proposed and demonstrated as early as 1993 as an attempt to minimize human errors resulting from inconsistent standards and calibrations of various medical instrument vendors [11]. Also, ventilator management with an integrated knowledge-based technology has been proposed [12]. Most automated monitoring studies tend to give heart monitoring more emphasis on vital state monitoring such as the object-oriented implementation demonstration to assess patient status using cardiovascular data [13].

There has been an identified need to utilize technology to better the care in a critical environment such as the critical-care centers, emergency and surgery rooms, or the intensive care units (ICU). Wireless PDAs have been demonstrated to carry ICU telemedicine [14] as well as home-based healthcare data [15]. The ICU monitoring networks have been developed to an advanced level of web-based representation [16] with emphasis on protocol design for such applications [17]. The wearable technologies are also very attractive in on-the-fly monitoring of patient health care [18]. However, they are limited in scope of what can be monitored and the relevance of monitored data to the patient's diseases. Home-based systems also make use of the wireless technologies in their monitoring systems with a limited access to only RS-232 serial interfaced medical instruments [19]. However, there is not a single system demonstration that would incorporate all possible medical instruments and administrative information into one information network.

Security of medical health records: Electronic medical record (EMR) systems are used for storing the patient's medical history and profile, data on patient care and management, and financial reimbursement information. The EMR system raised the issues of patient privacy and data security. The mistrust and insecurity created by the ease with which unauthorized persons can have access to sensitive medical information pushed Congress into enacting the Health Insurance Portability and Accountability Act of 1996 (HIPAA). HIPAA places strict compliance standards regarding the security of individual health information. Security and privacy are two indispensable aspects of all EMR systems. Security ensures that data are securely stored and transferred, and privacy ensures that data can be accessed only by the people who have authorization to view and use the data [20]. Attributes of information systems that play a key role in EMR security include authentication, authorization, integrity, accountability (including Non-repudiation), confidentiality, consent and audit ability [21-27]. Most security constructs for EMRs are based on a series of specifications regarding these attributes coupled with compliance to a set of laws and regulations governing the healthcare system (viz. HIPPA) [28]. A detailed discussion of the various EMR system designs is beyond the scope of this study and is not presented here. HIPAA mandates covered entities to implement policies and procedures to safeguard an individual's EMRs whilst in transmission, at rest, and in storage. In this study, we focus on the security of stored medical records, such as those obtained from networked medical devices as described earlier. Database servers with stored records are particularly prone to security attacks. Since a database contains most of the demographic, insurance and identity information of a patient, it is essential to secure it from attacks. We implemented open source tools [29] for National Security Agency (NSA) defined IAM and IEM methodologies to assess and evaluate the risks a database server housing medical information is subject to.

The rest of this paper is organized as follows: section 3 is on the networking issues and the proposed networking solution to medical instrument networking; section 4 is on the security issues for access to such medical instrument and patient examination data; section 5 summarizes our conclusions.

3. NETWORK OF MEDICAL INSTRUMENTS

Medical professionals need to be present at the location of all medical instruments such as at the bedside of a critical care patient in order to capture vital information. In many cases, medical professionals may have to manually control and adjust parameters of an instrument according to the measurements from a different instrument. Statistical research on personalized care is hard to execute when data are scattered and access is limited. Capturing and entering patient data is still a tedious, time-consuming, and error-prone process even though the information has to be real-time and potentially very sensitive for the well-being of the patient. Patient care itself is prone to errors because of error-prone manual recording of instrument measurements. There is a need for a common standard which allows for inter-networking of medical devices from different manufacturers.

A. Networking of Critical Care Medical Instruments

In 2000, CEN, ISO and IEEE joined to build a single set of standards called ISO/IEEE 11073 for point-of-care device communications to unify the interfaces of all medical devices [30, 31, 32]. Two of these five 11073 standards, ISO/IEEE 11073-30200 (cable-connected) and ISO/IEEE 11073-30300 (infrared-wireless), provide communication services and protocol definitions, consistent with IrDA (Infrared Data Association) specifications which are adapted as appropriate for ISO/IEEE 11073 applications. However, ISO/IEEE 11073 has not been able to generate a meaningful adoption by the industry. With advances in network communications technology, many researchers have been trying to connect isolated bedside medical instruments into a network. Most manufacturers have developed their own proprietary solutions failing to gain general acceptance.

MediCAN™ technology suite creates the interfacing hardware and related communication protocol in an open standard fashion for instruments to network in any healthcare environment. MediCAN™ system works towards a similar goal as ISO/IEEE 11073 [33, 34, 35] in being a candidate to become an open standard. MediCAN™ addresses communication services and protocol definitions based on Control Area Network (CAN) communication. MediCAN™ uses instrument adaptors and networking equipment to connect instruments on a CAN bus and then to a network, such as the Ethernet.

(i) In IEEE 11073, both wired and wireless versions are intended to provide communication between medical devices and external computer systems with plug-and-play and interoperable interfaces. Based on IrDA specifications, the connection link between Device Communications Controller (DCC) and Bedside Communications Controller (BCC) is a half-duplex, point-to-point communication.

Topology and Network: BCC or primary node is a hub connecting a local or remote external site via LAN or WAN. DCC or secondary node connects each medical

instrument to the communication network via BCC. Each device has a DCC and a BCC connecting to a gateway [30]. As shown in Fig. 1, adapter modules (DCC) together with the bedside modules (BCC) create the interfacing of the medical instruments to the network.

Infrared Link Access Protocol (IrLAP): BCC as a primary node would initiate a transaction such as device discovery or link negotiation. The secondary node (DCC) responds when spoken to by the primary. IrLAP consists of four phases: Device Discovery, Link Negotiation and Connection Establishment, Information Exchange, and Disconnection. Every frame has an address, a control field, and the payload.

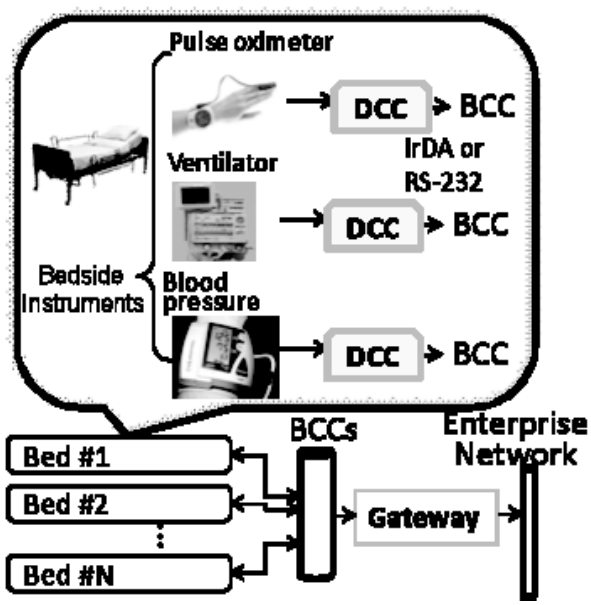


Figure 1. IEEE 11073 provides interfaces of instruments to the network through DCC-BCC (device and bedside communication controller) pairs. Implementations of this architecture have been limited to single PC demonstrations. The communication is based on IrDA and RS-232.

(ii) MediCAN™ Control Protocol (MCP) is the protocol layered over on ISO 11898:1995 CAN 2.0B (Control Area Network) defining physical layer and data link layer used for integrating medical instruments into the MediCAN™ System. MCP provides plug-and-play capability. Proposed MediCAN™ system is illustrated in Fig. 2 with one adaptor per instrument interfacing with a tightly-coupled CAN-based network for each set of bedside instruments.

Topology and Network: MCP allows direct communication only between a primary node called the Gateway (GNode) and one or more Device Nodes (DNodes) using a shared broadcast CAN bus as shown in Fig. 2. A DNode is the local access point for one or more medical devices. According to CAN 2.0B, the maximum data rate of a 40-

meter-long bus is 1 Mbps. GNode is a system providing an access point between a medical instrument network (using MCP) and an Ethernet network (using the MediCAN™ Gateway Protocol - MGP) over UDP.

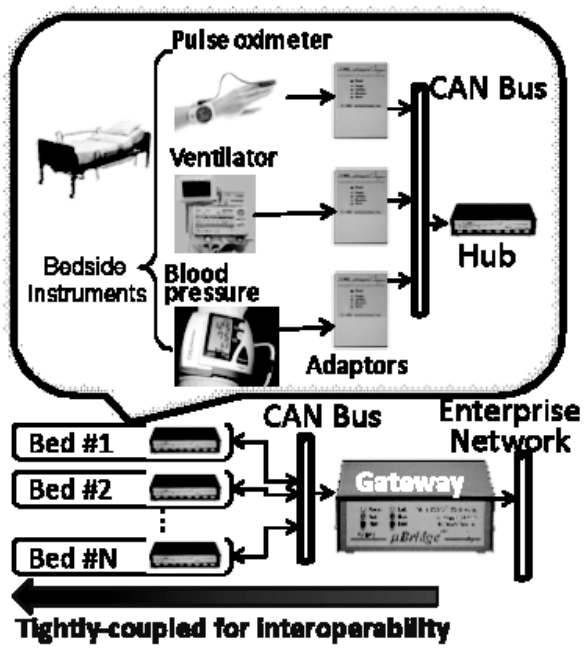


Figure 2. MediCAN™ system is composed of instrument adaptors at the bedside connected to a hub and then with other beds to a gateway to be accessed by users of the enterprise network.

Protocol Stack: Physical layer defines how signals are actually transmitted and deals with the description of Bit Timing, Bit Encoding, and Synchronization. CAN is a carrier-sense multiple-access protocol with collision detection and arbitration on message priority (CSMA/CD+AMP). With CSMA, each node on a bus must wait for a prescribed period of inactivity before attempting to send a message, and with CD+AMP, collisions are resolved through a bit-wise arbitration based on priority of each message in the identifier field of a message: the higher priority always wins the bus access.

B. Tightly-Coupled Network at the Bedside

The simulation of MCP was realized with three DNodes and one GNode on the CAN bus using discrete-event simulation approach. Measurement of the latency on frames transmitted between three DNodes and the GNode has been achieved. In particular, the latency only means the delay times any DNode would experience while waiting for the CAN bus to be idle before transmitting their frames.

(i) Simulation Assumptions

The simulation utilized the CAN bus with constant data rate of 1 Mbps with no propagation delay and processing delay on any point. There are three DNodes: $DNode_1$, $DNode_2$, and $DNode_3$, assumed that they all have passed Connect/Grant phase and already assigned an unique address from the GNode. $DNode_1$ always transmits constant bit rate data frames with the first frame randomly generated based on the exponential distribution. $DNode_1$ is simulating a streaming medical instrument such as a pulse oximeter's plethysmograph output. Streaming outputs involve a constant sampling rate signal to be sent at a constant bit rate. The ultimate goal of the simulation is to analyze the latency experienced by an event (e.g., a low heart rate) reported through a lower priority medical instrument while a higher priority medical instrument is continuously transmitting on the shared link.

$DNode_1$, $DNode_2$, and $DNode_3$ always transmit random data frames based on exponential distribution with $DNode_2$ having a higher priority on the bus than $DNode_3$. According to MCP frame types, $DNode_1$ is assumed to transmit report frames, and $DNode_2$ and $DNode_3$ to transmit response frames in function calls. Therefore, $DNode_1$ inherently has the most priority on the bus based on Typ field at the beginning of each frame followed by $DNode_2$, and $DNode_3$. In addition, for any function calls, the simulation assumed that there was no frame direction from GNode to either $DNode_2$ or $DNode_3$ and, certainly based on CAN communication, there was no frame transmission amongst the nodes $DNode_1$, $DNode_2$ and $DNode_3$. Only frames from $DNode_1$, $DNode_2$ and $DNode_3$ to GNode are implemented. At any point of time, all the DNodes have equal bit rates. Each MCP frame contains eight data bytes with total of 128 bits per frame based on CAN2.0B specification. Each simulation was run for 1 to 2.5 second, with each measurement repeated ten times. Data reported is the mean of these ten values.

There are two conditions of contention and retransmission:

- If a node tries to transmit when one of the other nodes is already transmitting the frame, the former has to wait until the latter finishes transmission plus inter-frame space interval.
- If a lower-priority DNode is about to send its frame at the same time a higher priority DNode tries to transmit, the higher priority node will be able to send its frame immediately and the lower-priority DNode has to stop and wait to re-send the entire frame until the higher-priority DNode is done transmitting plus an inter-frame interval.

(ii) Simulation Results

Simulation 1: A streaming medical instrument (e.g., $DNode_1$ represents a continuous measurement of oxygen saturation in blood) and another instrument (e.g., $DNode_2$ represents a temperature monitor) with event reporting capability. This experiment was designed for measuring the average delay of frames from $DNode_1$ (report data) with variable utilization of $DNode_2$ (function call data) and constant bit rate at 0.1 Mbps.

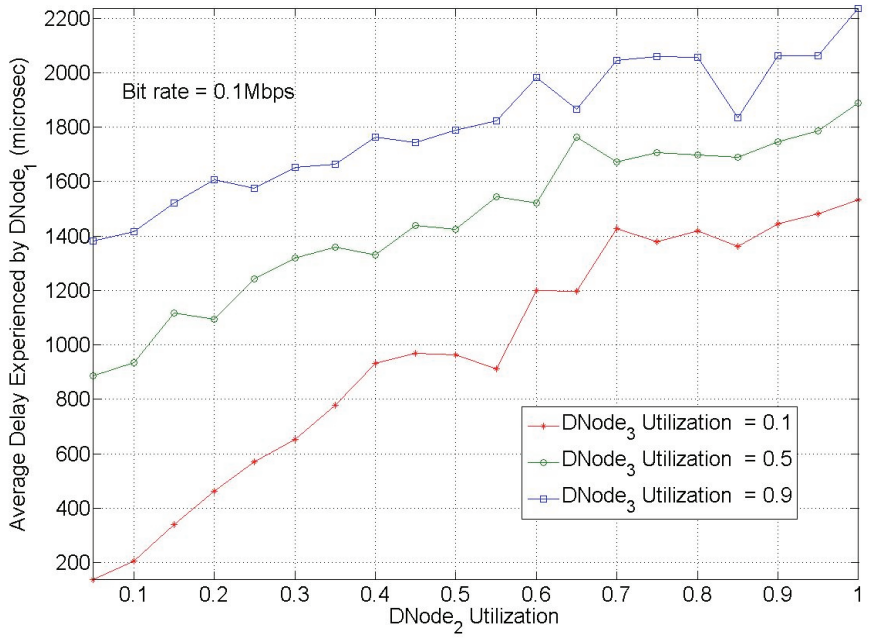
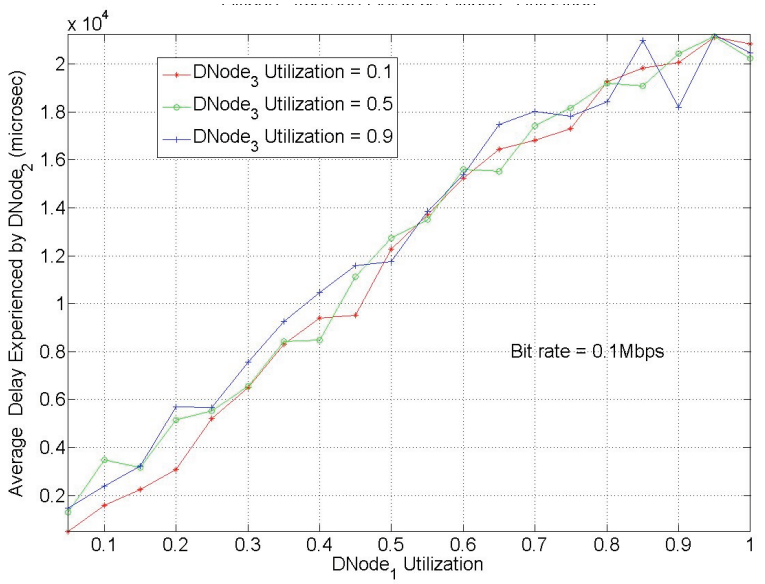


Figure 3. Average delay of DNode1 vs. utilization of DNode2

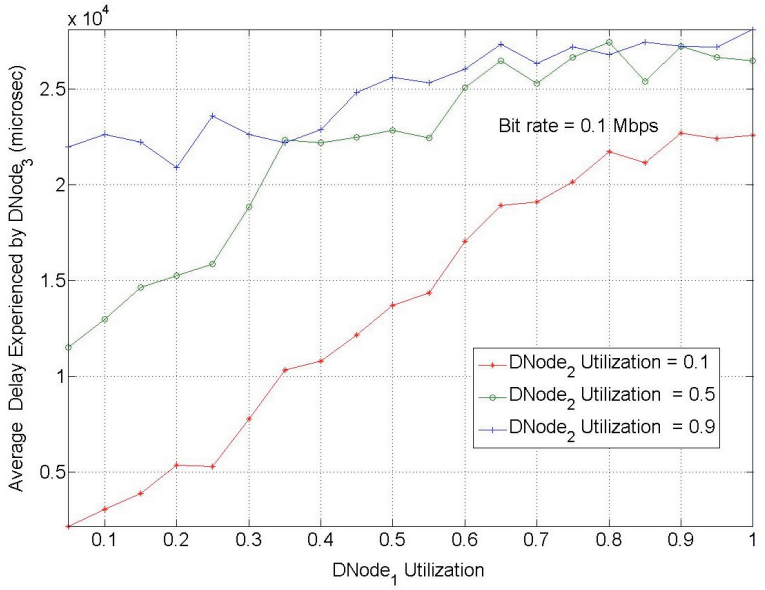
As shown in Fig. 3, as utilization of $DNode_2$ increases (as for example, when more events are reported due to an emergency), the delay experienced by the highest priority medical instrument ($DNode_1$ with streaming continuous data) also increases – from 0.2 msec to 2.2 msec. The increase in utilization of the lower priority node causes an increase in delay of the streaming nodes packets. However, the delay of the streaming node (highest priority node) is bounded for increasing rates of data streams in the CAN network. For a given frequency of emergency data reports, the streaming of the high priority medical data will experience a deterministic range of delay. For example, if $DNode_1$ has a pulseoximeter, when emergency traffic on a low priority temperature monitor on $DNode_2$ is increased, additional delays would be added to $DNode_1$'s reports in a bounded manner.

Simulation 2: Considering the same instrument assignments as in simulation 1, this experiment was designed for measuring the average delay of frames from $DNode_2$ and $DNode_3$ (as the lower priority nodes on the bus) with several bus utilizations (0.1, 0.5, and 0.9) and variable utilization for $DNode_1$. In this scenario, as the utilization of the link increases, frame rate also increases. Therefore, more bus arbitrations are won by $DNode_1$ resulting in more frame delays for the lower priority nodes. This explains why increased utilization (from 0.1 up to 1) for each of $DNode_2$ and $DNode_3$ caused increased delay (from ~2 msec – ~20 msec) in Fig. 4 (a) and (b), respectively. However, since increase in utilization causes the frame duration to shrink for all nodes, average delay times stay flat for each utilization value in $DNode_3$ case in Fig. 4 (b). Therefore,

lower priority traffic experiences somewhat flat delay for the range of streaming rates of another higher priority medical instrument.



(a)



(b)

Figure 4. (a) DNode2's average delay vs. DNode1's bit rate (For DNode3's utilization = 0.1, 0.5 and 0.9)
(b) DNode3's Average delay vs. DNode1utilization (For DNode2's utilization = 0.1, 0.5 and 0.9)

This simulation demonstrates the strength of a tightly-coupled system: **a bounded delay for event reports,** which provides predictability, as critical information would be reported through events, and is a desirable attribute for medical instrument networks for critical care. For example, if $DNode_1$ has a pulseoximeter and the monitoring rate is increased, this would not incur any additional delays for a low priority temperature monitor on $DNode_3$.

The simulations above demonstrate key features of the MediCAN™ protocol that would be beneficial for multiple devices networked in the intensive care environment. For example, simulation 1 suggests that increased utilization on $DNode_2$ can increase the delay experienced by $DNode_1$ (hosting a high priority device). This would allow a device reporting an emergency, such as an abnormal temperature to increase its utilization of the bus, while a streaming device is delayed – but, only in a bounded fashion. Critical monitoring needs of ICU would benefit from somewhat predictable delay expectations.

The data from the medical devices interfaced on the network can be automatically stored in the database for use as electronic medical records. In the following section, we briefly discuss security of stored data, and present a framework for evaluation of security constructs.

4. SECURITY CONSIDERATIONS

The responsibility of protecting health information stored in a database is critical for security and privacy management. Access to EMR should be granted on a strict need-to-know basis. The scope of this case study was to identify, assess and evaluate risks faced by a database server. To demonstrate the use of open source tools and NSA triad for HIPAA compliance, a server hosting a web application with a Structured Query Language (SQL) database backend was used. The server specifications are as follows: Operating System; Microsoft Windows 2008 server, Database; SQL Server 2005 and Web Application Server; IIS7.0.

NSA's triad for information security (INFOSEC) consisting of INFOSEC Assessment methodology (IAM) [36], INFOSEC Evaluation methodology (IEM) [37] and Red Teaming methodologies were employed to demonstrate a process that HIPAA covered entities may use to assess their policies and procedures, scan for vulnerabilities, and check for the level of difficulty of their exploitation [28].

Phase I - INFOSEC Assessment Methodology (IAM)

The assessment phase involves (1) review of medical information security policies, procedures and practices already in place, and (2) defining critical medical information, and creating an information criticality matrix. Details of the policies and procedures are dependent on the system, institutional and application requirements. We adopted the security policies and procedures mandated by the Information Technology Office at the University of Houston. Components of the critical medical information related to patient privacy include data on identity information [21], health information, suggested treatment, and login credentials. After identifying medical information critical for our application, the information criticality matrix shown in Table 1 was defined.

Table 1. Information Criticality Matrix. Critical information is classified as high, medium and low based on the financial and reputational losses faced by the covered entity in the event this information is lost or misused

Information	Confidentiality	Integrity	Availability
Patient Identity Information	High	Medium	Medium
Health Information	High	High	High
Medical records	High	High	Low
Suggested treatment	High	High	Low
Login credentials	High	Medium	Low
Employee information	Medium	Low	Low

Critical information was classified as high, medium and low based on the financial and reputational losses faced by the covered entity in the event this information is lost or misused. A high classification includes any major breach and fines (>$50,000), disclosure of medical records and loss of clients. A medium classification includes fines >$1000 and <$50,000, and loss of employee information, while the low classification includes delays in accessing the information or business operations without any financial repercussions.

Phase II - INFOSEC Evaluation Methodology

The INFOSEC evaluation phase made use of Nmap and Nessus to identify open ports, services running and vulnerabilities that could be exploited. The first test (Nmap) involved fingerprinting the Operating System (OS) and checking for open ports. The second test (Nessus) was used to identify vulnerabilities and to quantify risk faced by the health care provider if the vulnerabilities were exploited. Network Mapper (Nmap) [xvii] is an open source tool used to discover systems and services running on a network. It is capable of detecting the operating system, uptime, the software used to run a service, its version number, and providing a "traceroute". This tool can be used by a security administrator or hacker to search for, and identify vulnerable open ports on servers. Quite often hackers use Nmap to identify vulnerable systems and gain unauthorized access. Details on implementing the NMap tools are not presented [36]. Nmap gives away a lot of information about the scanned machine: open ports, services running, the OS. An attacker could try to gain unauthorized access by connecting to one of the open ports and exploiting an unpatched vulnerability. In our testing, NMap results, indicated that the machine scanned was a Microsoft Server with an open port 1433, which indicated that the SQL Server was listening on this port. With this knowledge, we could then use Nessus to dig deeper and look for vulnerabilities that could be exploited and needed to be fixed.

Nessus is a vulnerability scanner that scans for and provides a list of vulnerabilities that could be exploited [36]. With a Graphical User Interface, the open source version of this tool can prove to be very helpful for both a security administrator looking to fix holes and a hacker looking to exploit them. It can scan a single host or a list of hosts and sort the results by vulnerabilities identified and CVE (Common Vulnerabilities and

Exposures) numbers. Information about the Common Vulnerabilities and Exposures can be obtained from http://cve.mitre.org/cve/. Each CVE is given a rating by the Common Vulnerability Scoring System (CVSS) [xviii]. The risk rating for each of the CVEs identified was obtained from the National Vulnerability database and a vulnerability matrix was created to assess the amount of risk our organization may face if any of these vulnerabilities is exploited [37]. This matrix was built on two axis points; (1) the Information Criticality/Priority defined during Assessment process (shown in Table 1), and (2) the Vulnerability Priority as defined by industry standard ratings for each technical finding of CVE – Common Vulnerabilities and Exposures.

Table 2. Vulnerability Weight Matrix.

	High	**Medium**	**Low**
Critical Impact Weight	3	2	1
Vulnerability Weight	6	4	2
Vulnerability Matrix	8 to 9	5 to 7	3 to 4

The scores for each block in the matrix are then determined based on the industry ratings for the found vulnerabilities, input from the IAM about the criticality of each information type, and the opinion of the evaluator on the actual impact of the vulnerability based on expertise. Using this procedure, and based on the Common Vulnerability Scoring System (CVSS) severity rating, we determined the vulnerability matrix for our test server (see Table 2).

For the test server in this study built based on the networking and storing of medical instrument and patient data, the vulnerability matrix in Table 3 was created

Table 3. Vulnerability ratings identified for a patient information database server.

Vulnerability	**Finding number**	**Severity**	**Patient Identifiable Information** *Confidentiality/ Integrity/ Availability*	**Health Information** *Confidentiality/ Integrity/ Availability*	**Medical Records** *Confidentiality/ Integrity/ Availability*	**Suggested Treatment** *Confidentiality/ Integrity/ Availability*	**Login Credentials** *Confidentiality/ Integrity/ Availability*	**Employee Information** *Confidentiality/ Integrity/ Availability*
CVE 2005-3595	1	High	9/8/8	9/9/9	9/9/7	9/9/7	9/9/7	8/7/7
CVE 2000-0222	2	High	9/8/8	9/9/9	9/9/7	9/9/7	9/9/7	8/7/7
CVE 1999-0504	3	High	9/8/8	9/9/9	9/9/7	9/9/7	9/9/7	8/7/7
CVE 1999-0506	4	High	9/8/8	9/9/9	9/9/7	9/9/7	9/9/7	8/7/7
CVE 1999-0505	5	High	9/8/8	9/9/9	9/9/7	9/9/7	9/9/7	8/7/7
CVE 2002-1117	6	Medium	7/6/6	7/7/7	7/7/5	7/7/5	7/6/5	6/5/5

Table 3 lists the following Common Vulnerabilities and Exposures that were identified:

- CVE-2005-3595: blank password for the Administrator account, CVSS Severity: 10.0 (HIGH).

- CVE-2000-0222: inactive Administrator password until the system has rebooted, CVSS Severity: 10.0 (HIGH).
- CVE-1999-0504: administrator account has a default, null, blank, or missing password, CVSS Severity: 7.5 (HIGH).
- CVE-1999-0506: administrator account has a default, null, blank, or missing password, CVSS Severity: 7.2 (HIGH).
- CVE-1999-0505: administrator account has a guessable password, CVSS Severity: 7.2 (HIGH).
- CVE-2002-1117: possible anonymous listing of the SAM database and shares, CVSS Severity: 5.0 (MEDIUM).

Phase III – Red Teaming Methodology

The last phase of the NSA Triad, Red Teaming, involves conducting penetration tests to gain unauthorized access to the test server. Most data storage servers are networked to allow accessibility through the internet. The four most common attacks to compromise a database server include direct connection, password cracking, SQL Injection and direct exploit [38]. First, an attacker can try to directly connect to a SQL server without firewall protection. Thus, placing database servers behind a firewall is the first line of defense against their compromises. The attacker can easily identify, scan for and establish a connection with naked servers using Nmap and SQLPing (tool used to (i) reveal information such as details about all named instances installed on a server prior to connection, and (ii) send discovery packets to entire networks for mass interrogation). Second, via password cracking a hacker can compromise all of the data stored in the database. The attacker can attempt to look for and crack weak passwords used for logging in to the SQL Server. SQL Servers configured with a default password are most prone to this kind of attack. Additionally, weak user passwords for SQL Server authentication mode are prone to brutal force and dictionary attacks using any freely available password cracking tools [39]. Third, via SQL Injection, an attacker or a malicious insider compromises the application that provides the interface for database access, and uses SQL statements to edit, delete or update the patient data. The attacker could also modify stored procedures to create new system logins. He/She could 'watch' the database and have access to any patient data entered in the database. SQL Injection attacks exploit the vulnerabilities in query text such as the single quote apostrophe that is used as a string delimiter, the line comment identifier of the double hyphen, the semicolon used to delimit separate SQL queries on one line, the union operator to append data from other tables to the originally intended result set, and the ability to find out database settings, table names and column information [40]. Finally, a hacker or an insider with a malicious intent can directly exploit servers by making use of tools like Nessus or Microsoft Baseline Security Analyzer (MBSA) to look for database and underlying OS vulnerabilities. Unpatched servers found can be exploited by a hacker using a penetration testing tool like Metasploit's msfconsole. Metasploit can be downloaded directly from the Metasploit website [41]. Thus in the Red Teaming Phase, we used SQLPing, Metasploit and SQL Injection methodologies to check if the database could be exploited. Table 4 lists the risks identified during the preliminary testing of our system, and outlines strategies to address the risks.

Table 4. List of risks identified using the NSA triad assessment and mitigating strategies.

RISK	MITIGATION
Direct connection	Protect the server with a firewall and block unauthorized connections on the default transmission control protocol (tcp) port.
Weak passwords vulnerable to brute force and dictionary attacks	Implement a strong password policy. Use mixed or windows authentication when possible.
Direct exploit attacks	Regularly patch servers; critical updates should be implemented immediately after release. Enable only the services required to get the job done.
SQL Injection	Validate user input. Examples: (1) A single quote should be replaced with double quotes. (2) Reject two hyphens which will comment out any query appended after the hyphens. (3) Limit number of characters in a textbox. (4) Avoid building dynamic queries. (5) Limit database object permissions.
Unauthorized changes to the database	Regularly monitor database logs to check for unauthorized access. Users should not be allowed to delete or insert new tables. Access to the database should strictly follow the principle of least privilege.

General SQL server 2005 security best practices [42] can help covered entities securely configure, manage and audit their database servers. With a little effort, the tools described in this study can successfully depict an organization's security posture and help fix vulnerabilities discovered.

5. CONCLUSION

We have presented security overview and network architectural considerations for medical instrument networking in critical care. Network architectures are presented with a comparison between a proposed standard of IEEE 11073 suite of standards and a tightly-coupled case implemented with MediCAN™. The comparison showed strengths of the standard as a unified object-oriented information model with high overhead in the networking architecture. On the other hand, the tightly-coupled system showed less overhead on the network with strengths in event/alarm handling while a stream of monitoring data is present on the shared link. We have implemented such a patient information server with storage of medical data in a database. According to HIPAA, any risk associated with these stored data records needs to be identified, quantified, mitigated and monitored. In this paper, we proposed the use of NSA triad and open source tools for health care providers to optimize HIPAA compliance of medical data in storage. Although there are numerous other open source tools, we implemented a set of tools that are considered to be the most capable and ranked among the top 10 security tools. The INFOSEC Assessment phase defined critical medical

information, and proposed and assessed HIPAA security policies and procedures that could be a good fit for the test server. The INFOSEC Evaluation phase made use of Nmap and Nessus to identify open ports, services running and vulnerabilities that could be exploited. The last phase, Red Teaming, involved the use of SQLPing, Metasploit and SQL Injection to check if these could exploit the database. We also utilized the general SQL server 2005 security best practices to securely configure, manage and audit our database server. The NSA methodology was implemented to optimize HIPAA compliance for data in storage. Open source tools were successfully used with the proposed NSA triad methodology to assess and evaluate risks associated with medical data in storage. Another topic that may be addressed in future projects is the security of medical data during wireless communications.

ACKNOWLEDGMENT

This work was supported through equipment donations for evaluation purposes by MediCAN™ Systems Inc. and through maintenance and setup of the patient database server by Information Technology Department, College of Technology, University of Houston.

LIST OF ACRONYMS

API:	Application Programming Interface
ARQ:	Automatic Repeat Request
CAN:	Control Area Network
CRC:	Cyclic Redundancy Check
CVE:	Common Vulnerabilities Exposures
CVSS:	Common Vulnerability Scoring System
CSMA/CD+AMP:	Carrier Sense Multiple Access/Collision Detection + Arbitration on Message Priority
DCC/BCC:	Device/Bedside Communication Controller (from IEEE 11073)
DNode/GNode:	Device/Gateway Node
EMR:	Electronic Medical Record
GUID:	Global Unique Identification
HIPAA:	Health Insurance Portability and Accountability Act
IAM:	INFOSEC Assessment Methodology
ICU:	Intensive Care Unit
IEM:	INFOSEC Evaluation Methodology
INFOSEC:	Information Security
IrLAP:	Infrared Link Access Protocol
LAN/WAN:	Local/Wide Area Network
MCP:	MediCAN Control Protocol
MediCAN:	Proposed technology suite by MediCAN Systems Inc. for medical instrument networking on the CAN bus in an interoperable fashion.
NSA:	National Security Agency
PDA:	Personal Digital Assistant
RF:	Radio Frequency

SQL:	Structured Query Language
TCP:	Transmission Control Protocol
UDP:	User Datagram Protocol

REFERENCES

[1] F. A. Mora, G. Carrault, and J.-P. Le Pichon, Intelligent patient monitoring and management systems: a review, *IEEE Engineering in Medicine and Biology*, 1993, Dec.

[2] D. Panescu, Emerging technologies: healthcare applications of RF identification, *IEEE Engineering in Medicine and Biology Magazine*, 2006, May/June.

[3] E. J. Manders, and B. M. Dawant, Data acquisition for an intelligent bedside monitoring system, 18th Intern. Conf. of IEEE Engineering in Medicine and Biology Society, 1996.

[4] B. M. Dawant, S. Uckun, E. J. Manders, and D. P. Lindstrom, The SIMON project: model-based signal acquisition, analysis, and interpretation in intelligent patient monitoring, IEEE Engineering in Medicine and Biology, 1993, Dec..

[5] S. Warren, J. Yao, R. Schmitz, and J. Lebak, Reconfigurable point-of-care systems designed with interoperability standards, Proceedings of the 26th Annual International Conference of the IEEE EMBS, 2004.

[6] J. Yao, R. Schmitz, and S. Warren, A wearable point-of-care system for home use that incorporates plug-and-play and wireless standards, *IEEE Transactions on Information Technology in Biomedicine*, 2005, 9(3) Sept.

[7] P. Varady, Z. Benyo, and B. Benyo, An open architecture patient monitoring system using standard technologies, *IEEE Transactions on Information Technology in Biomedicine*, 2002, 6(1) March.

[8] M. F. Freeman – chair of nomenclature maintenance agency, The global medical devices nomenclature: a summary, 2002.

[9] S. Price, R. Summers, and D. J. Williams, Medical device databases: a scoping study, Proceedings of the IEEE Engineering in Medicine and Biology, 2005.

[10] A. Prentza, S. Maglavera, N. Maglaveras, and D. Koutsouris, Healthcare services towards individualized wellness (i-wellness), Proceedings of the IEEE Engineering in Medicine and Biology, 2005.

[11] R. C. Watt, E. S. Maslana, and K. C. Mylrea, Alarms and anesthesia: challenges in design of intelligent systems for patient monitoring, IEEE Engineering in Medicine and Biology, Dec. 1993.

[12] R. Summers, E. R. Carson, and D. G. Cramp, Ventilator management: the role of knowledge-based technology, IEEE Engineering in Medicine and Biology, 1993, Dec.

[13] T. Sukuvaara, M. Sydänmaa, H. Nieminen, Arno, Heikelä, and E. M. J. Koski, Object-oriented implementation of an architecture for patient monitoring, IEEE Engineering in Medicine and Biology, 1993, Dec.

[14] F. Lamberti and B. Montrucchio, Ubiquitous real-time monitoring of critical-care patients in intensive care units, Proc. of the 4th IEEE Conf. on Information Technology Applications in Biomedicine, 2003.

[15] D. Salamon, M. Grigoni, M. Gianni, M. Liberti, S. De Luca, A. Bei, and G. D'Inzeo, Indoor telemedicine in hospital: a PDA-based flexible solution for wireless monitoring and database integration, Proceedings of the IEEE Engineering in Medicine and Biology, 2005.

[16] João Bosco da Mota Alves, Juarez Bento da Silva and Suenoni Paladini, A low cost model for patient monitoring in intensive care unit using a micro web-server, IADIS Virtual Multi Conference on Computer Science and Information Systems, MCCSIS 2006.

[17] I. Niubo, M. Mulet, T. Gual, and A. Rodriguez, Designing a communication protocol for a central station monitoring system, Proceedings of the 25th Annual International Conference of the IEEE EMBS, 2003.

[18] B. Wu, Y. Zhou, X. Zhu, Q. Yan, L. Zhu, and G. Li, A novel mobile ECG telemonitoring system, Proceedings of the IEEE Engineering in Medicine and Biology, 2005.

[19] J. W. Seo, M. S. Ryu, K. S. Park, and D-U. Jeong, A home-based bedside monitoring system of ECG via Bluetooth protocol, IEEE, 2003.

[20] D. Daglish and N. Archer, Electronic Personal Health Record Systems: A Brief Review of Privacy, Security, and Architectural Issues, Congress, pp.110-120, 2009 World Congress on Privacy, Security, Trust and the Management of e-Business, 2009.

[21] ANSI, ISO/TR 18308 Health Informatics – Electronic Record Architecture, ISO 2003.

[22] J. Grimson, W. Grimson, D. Berry, G. Stephens, E. Felton, D. Kalra, P. Toussaint, O.W.Weier, A CORBA-based integration of distributed electronic healthcare records using the synapses approach, IEEE Trans. *Inf. Technol. Biomed*. 1998, 2, 124–138.

[23] A. Shabo, A global socio-economic-medico-legal model for the sustainability of longitudinal electronic health records. Part 2, Methods Inf. *Med*. 2006, 45, 498–505.

[24] C. Lovis, S. Spahni, N. Cassoni, A. Geissbuhler, Comprehensive management of the access to the electronic patient record: towards trans-institutional networks, Int. J. *Med*. Inform. 2007, 76, 466–470.

[25] A.R. Bakker, The evolution of Health Information Systems, security in practice and open issues, Stud. *Health Technol. Inform*. 2003, 96, 15–20.

[26] B. Blobel, Advanced and secure architectural EHR approaches, *Int. J. Med. Inform*. 2006, 75, 185–190.

[27] Helma van der Lindena, Dipak Kalrab, Arie Hasmanc, Jan Talmona. Inter-organizational future proof EHR systems: A review of the security and privacy related issues. *International journal of medical informatics*, 2009, 141–160.

[28] Massey, AK, Otto, PN, Hayward LJ, Anton, AI. Evaluating existing security and privacy requirements for legal compliance, *Requirements Eng*, 2010, 15, 119-137.

[29] H. Farooqui, E. Crowley, D. Gurkan, and F. Merchant, Open Source tools for optimizing HIPPA compliance, Biomedical Society Annual Conference, 2009, Oct.

[30] M. Gallaraga, et.al., Proposal of an ISO/IEEE 11073 platform for healthcare telemonitoring: plug-and-play solution with new use cases, 29th IEEE EMBS Annual International Conference, 2007.

[31] S. Warren, J. Yao, R. Schmitz, and J. Lebak, Reconfigurable point-of-care systems designed with interoperability standards, *IEEE EMBC* , 2004, 26, 3270 – 3273.

[32] IEEE Standards Association page: http://standards.ieee.org/

[33] R. J. Kennelly and R. M. Gardner, Perspectives on development of IEEE 1073: the medical information bus (MIB) standard, *Int. Journ. Clin. Monit. Comput*., 1997, 14, 143-149.

[34] P. K. McKneely, F. Chapman, and D. Gurkan, Plug-and-Play and Network-Capable Medical Instrumentation and Database with a Complete Healthcare Technology Suite: MediCAN, 2007 HCMDSS and Medical Device Plug-and-Play Interoperability.

[35] Suman Gumudavelli, Paul K. McKneely, Pongnarin Thongpithoonrat, D. Gurkan, Frank M. Chapman, "Medical Instrument Data Exchange, IEEE EMBC 2008, 2008, August.

[36] Insecure.Org. Top 100 Network Security Tools. Retrieved May 2009, from http://sectools.org/: http://sectools.org/

[37] Homeland Security. (N.A, N.A). National Vulnerability Database. Retrieved May 2009, from cve.mitre.org: http://cve.mitre.org/cve/

[38] System Administration, Networking, and Security Institute. (2007). Top 20 Internet Security Problems, Threats and Risks. Retrieved July 2009, from www.sans.org: http://www.sans.org/top20/#s7

[39] Grindlay, B., & Litchfield, D. (2005). Database Hacker's Handbook: Defending Database Servers. Wiley, John & Sons, Incorporated

[40] Microsoft. SQL Injection. Retrieved July 2009, from msdn.microsoft.com: http://msdn.microsoft.com/en-us/library/ms161953.aspx

[41] Rapid7. (N.A, N.A). Dowload the Metaspoit Framework. Retrieved September 2009, from www.metasploit.com: http://www.metasploit.com/framework/download/

[42] Bob Beauchemin, SQL Server 2005 Security Best Practices - Operational and Administrative Tasks, March 2007. Retrieved July 2010 from http://download.microsoft.com/download/8/5/e/85eea4fa-b3bb-4426-97d0-7f7151b2011c/SQL2005SecBestPract.doc.