A Network Security Architecture to Reduce the Risk of Data Leakage for Health Care Organizations

Richard Rauscher
Col. of Medicine and Dept. of Computer Science & Eng.
Pennsylvania State University
Hershey, Pennsylvania, USA
rauscher@psu.edu

Raj Acharya
Department of Computer Science & Engineering
Pennsylvania State University
University Park, Pennsylvania, USA
acharya@cse.psu.edu

***Abstract*— Health care is a highly regulated industry in which much value is placed upon privacy and confidentiality. The business of health care, particularly in certain academic environments, requires access to data of varying sensitivities, including information from the public Internet. This paper proposes a VLAN-based architecture for segregating data of varying sensitivities, a list of components that facilitate access to and distillation of data, and a method for one-way promotion of individual nodes from areas of lower security to areas of higher security. The proposed solution is an implementable and pragmatic approach to reducing the risk of data leakage. Quality of experience (QoE) measures of two methods for access (node promotion and porthole-based access) are compared. The node promotion method improves the user-perceived responsiveness of applications over the porthole-based method while reducing flexibility.**

Keywords—health care information systems, electronic health records, VLAN, QoE, network security

I. INTRODUCTION

Health care is a highly regulated industry in which much value is placed upon privacy and confidentiality. The business of health care, particularly in certain academic environments, requires the use of data of varying sensitivities, including information from the public Internet. This paper proposes a VLAN-based architecture for segregating data of varying sensitivities, a list of components that facilitate access to and distillation of data, and a method for one-way promotion of individual nodes from areas of lower security to areas of higher security. The inspiration for this work was the authors' experience at several large academic health centers (AHCs) where the need to restrict access to confidential patient information was often challenged by the need to ensure the free flow of information required to cultivate a rich and collaborative research and educational environment. This research is supportive of future work which seeks to minimize the risk of data leakage while making use of hybrid computing clouds.

It is noteworthy that no system can prevent all *intentional* forms of data leakage. The proposed architecture does nothing to prevent egregious behavior by authorized individuals who are committed to acting unethically or illegally. For example, technical security won't prevent a bad actor from capturing a screen image of confidential data using a camera or smart phone.

This paper will detail: i) an implementable approach for managing data with varying degrees of sensitivity, and ii) a new method for dynamically changing VLAN assignments by specific nodes. *A note about wording:* we often refer to sensitive data metaphorically as a pollutant to be contained. This is apropos as sensitive data have many of the same characteristics as dangerous chemicals: they are useful if managed well but dangerous if control is lost.

II. BACKGROUND

The need to restrict the flow of confidential information is a fundamental component of information security. Data leakage is the unintentional flow of data from trusted systems and networks to less trusted systems and networks. There are daily accounts in the popular press in the United States about unintentional data leakage [1]. There are many examples of where patient data were inappropriately stored on unencrypted laptop computers, written to portable storage devices or displayed on public web sites. The Bell-LaPadula (BLP) model remains the authoritative standard reference model for multilevel security. BLP is a purist approach. It has been well-recognized that there are pragmatic needs that cannot be addressed in an environment that stringently adheres to BLP [2]. For example, a strict implementation of BLP in health care would prevent patients from accessing their own health information from their (presumed to be insecure) personal computers and devices. This may reduce patient engagement and would be contrary to efforts promoted by health providers and governmental agencies. Some data leakage management schemes have sought to classify every datum of every system and facilitate the management of leakage avoidance through novel programming language constructs and appropriate technical controls. The abundance of legacy applications and the slow rate of change of applications in health care settings [3] makes these largely academic efforts impractical to apply.

In the United States, the Health Insurance Portability and Accountability Act (HIPAA) defines broadly how health care data are to be managed and secured [4]. There are similar laws in many jurisdictions worldwide. Although HIPAA was created in 1995 and went into full effect in 2003, varying degrees of enforcement and penalties have impaired the effectiveness of its adoption. In 2009, as part of rapidly-enacted legislation designed to avoid economic disaster, the penalties associated with non-compliance of HIPAA were strengthened [5]. A 2011 study found significant variability among AHCs regarding compliance with HIPAA. Information security continued to be described as an afterthought [6]. Most AHCs surveyed lacked sufficient management support, culture and technical measures to ensure compliance. Patient data, though well-publicized due to

HIPAA, is not the only data classification of concern for AHCs. They must also be concerned with the privacy of student data (in the United States, this is codified in the FERPA rules [7]), payment card data (as codified in the PCI-DSS contractual requirements [8]) and other rules depending upon the regulatory or contractual framework that governs the data. Additionally, AHCs have a moral and ethical obligation to ensure the privacy of patients and research subjects against emerging threats. During the 1990s and early 2000s, genetic/genomic information that lacked specific identifiers was considered to be "de-identified". However, in 2013, researchers were able to successfully re-identify genomic data that was thought to be de-identified through the application of several external databases [9]. Individual data use agreements, formal and informal agreements to collaborate between institutions, individual scientists and physicians add further dynamism and thus complexity. Regulations, threats and relationships are changing rapidly. Additionally, users at AHCs often require access to multiple classifications of data as part of their workflow. A well-understood framework to reduce the risk of data leakage would be useful to the health care industry and specifically at AHCs.

Proprietary methods for virtually segregating local area network (LAN) traffic over switched link layer networks were introduced in the mid-1990s. The IEEE amended the 802.3 protocol in 1998 to officially establish a standard for VLAN traffic [10]. Early in the history of VLANs, there were discussions about using VLANs to segregate traffic based on policy [11]. There have been many examples of research and practical implementations that have focused on using VLANs to segregate data [12-14]. We have considered this previous research in our architecture.

III. Architecture of System

The architecture of this system is meant to be implementable using existing protocols with minimal modifications and existing applications. There are a number of important considerations that drive this solution:

- pragmatism is key -- this solution must be implementable using current technology and current (or old) applications;
- therefore, it is not practical to classify every datum in a system; systems will be classified based on the most sensitive data they store (this assumes that an ordering exists upon which the data in systems can be compared);
- highly sensitive data must be viewable with restrictions from low security areas; realistic needs of clinicians such as remote access to sensitive data must be satisfied; and
- "multiple use" devices must be able to transition from being classified as low sensitivity to high sensitivity dynamically; a method should exist to "reset" the device to low sensitivity.

The overall architecture functions using the constructs described in the following sections.

A. Network Zones

Each network zone has a specific characteristics: a security designation which describes which data may transit and be stored within it; membership requirements which must be met by any node connecting to the network and enforced through administrative or technical mechanisms; a set of privileges associated with the security designation; and a set of prohibitions. A practical example of this would be a network that permitted the storage and transit of regulated health data as described below in Table 1.

Table 1: Example of Zone Characteristics

Characteristic	**Example Restriction**
Security designation	PHI Zone
Membership requirements	Antivirus software; host-based IDS; 802.1x authentication
Privileges	Create, access, modify PHI
Prohibition	No Internet access; no access to email system

In this scheme, we assume that the network can securely segregate and maintain the separation of packets with different designations (tags). Modern layer two (switched Ethernet) networks perform this through the use of IEEE 802.1Q or other similar mechanisms. It is noteworthy that in our proposal there is no direct connectivity between zones of differing security classes. A layer three packet emanating from one zone *cannot* enter another zone. All information is conveyed through various filtering and proxying gateways at the application level of the network stack.

B. Sources

1) Static Sources

Packets emanate from information sources. At any time *t*, every information source *S* has a classification designation C(*S*). The packets originating from these sources are tagged with the same designation. Static sources have well-defined designations and may not change during their lifetime. An example of a static source may be a hospital registration system. It stores and computes upon protected health information (PHI) which has certain legal requirements which are enforced through technical measures.

2) Dynamic Sources

There are also dynamic sources of information. The security designation associated with dynamic sources may grow higher during operation but not lower. Formally, $C(S) = i \rightarrow C(S) = j$, where $j > i$. However, the source may not make the reverse transition without executing the "decontamination" process (see below). An example of a dynamic source system would be a general purpose workstation which, by default, is set to the lowest level of access S_0. A general purpose secure workstation may access insecure systems such as the Internet. It may also access secure systems such as the registration system in the previous example. It may not, however, access the insecure system *subsequent* to accessing the secure system. Permitting it to do so would create a path where data may have left the registration system, been recorded on the insecure workstation and then transmitted to a lower security system (and violate

the "no writes down" rule of BLP). Thus, before the system may receive a packet from of high security, its own security must be changed (in this case, $S_0 \rightarrow S_P$). Once the workstations designation is set to S_P, it may no longer send packets to targets with lower security designations.

C. Porthole

Portholes (as opposed to the over-used term "portal") are secure gateways that permit the access of higher security zones from lower or differing security zones. The portholes are designed to consolidate connectivity and minimize the risk footprint. Like their namesakes on ships, the porthole is/should be designed to provide an opening but not facilitate egress. Data may be viewed through a porthole but not copied (meaning that the risk of "copying and pasting" is removed). Practically and technically, however, the existence of the porthole increases the risk of data leakage (over having no access whatsoever) as data may still be intentionally leaked through screen capture software or simply by taking a picture of the screen.

D. Declassifiers

Declassifiers are secure data processing mechanisms that accept as input information with security designation S_i (within a zone capable of supporting data with characteristic S_i) and output data with security designation S_j where $S_i > S_j$. For example, a declassifier may take as input several identified patient records and output statistical information.

E. Sanitizers

Sanitizers are mechanisms and procedures that cleanse dynamic source nodes so that their security level and associated network zone may be "reset". The sanitizer "decontaminates" the node of any data that should not leave the high security zone. Practically, this typically translates to erasing the long-term storage, resetting the RAM and reinstalling the operating system and application software.

F. Example

Figure 1 below depicts an example design for protecting health data with different zones and different interface mechanisms in a static environment. In our example, we depict a user of a personal computer accessing data with a high security classification from a zone of lower security classification through portholes. We also depict the use of declassifiers and how they would make increasingly abstracted patient data available through i2b2 [15] within different zones.

G. Security Benefits and Risks

The proposed architecture reduces the possibility of data leakage through accidental disclosures such as copy-and-pasting, emailing and posting data into systems on the public Internet. Furthermore, the total lack of connectivity also protects against botnet-like leakages or other types of malware. Assuming that the network is and remains secure, a malware infection could corrupt data but not expose it outside of the organization (given the assumption that Internet connectivity is in the list of prohibitions for a secure zone).

The risk considerations and assumptions associated with the architecture are described below.

1) Porthole Image Capture

As described, the portholes access data through a secure mechanism of "screen scraping" which facilitates viewing of regulated data but not transmission. These protections can always be defeated through mechanisms that capture screen images (which could be something as simple as a camera) and the data could be re-constituted using optical character recognition techniques.

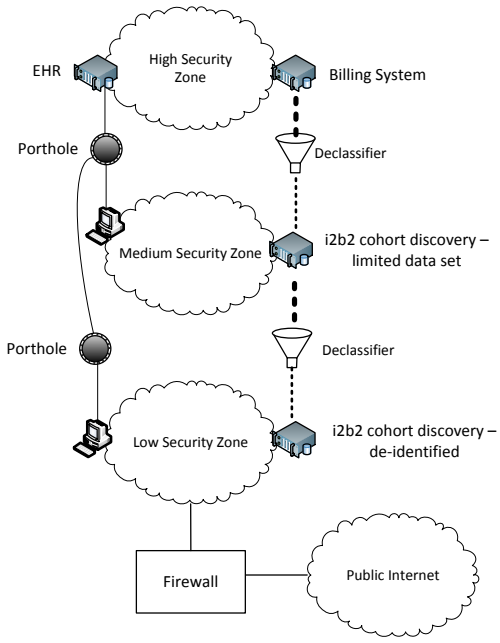

Figure 1: Data Leakage Protection Architecture

2) Declassifiers

The declassifiers which are used to aggregate statistics about data or otherwise reduce the security classification of certain data must be formally evaluated to ensure that sensitive information cannot be leaked. Organizational policies, mature data governance and rigorous testing routines are required to ensure that the declassifiers don't become a path for data leakage.

3) Network Isolation

A foundational assumption of this work is that network isolation is feasible and that VLAN hopping or other kinds of VLAN or network manipulation are improbable.

H. Performance Considerations

Performance concerns are the motivation behind the second contribution of this paper: a method to dynamically promote a node from a low security zone to a high security zone. We are concerned about the user-perceived performance of access to applications (also known as "quality of experience" or QoE). QoE is defined differently in several papers[16-18], so we will define it here: the user perceived experience associated with usage activities. We will concentrate on objective measurable events and leave user satisfaction for future work.

Based on previous experiences measuring the performance of EHRs [19] and work by Casas et al [18], we know that most user-triggered network transmissions involve keystrokes or mouse clicks while most server-triggered transmissions update screens. End-to-end delay between two nodes on a network is a function of the sum of the delays related to network queuing, transmission, propagation and processing time [20]. In the simplified mathematical model below, the QoE, is a function of the network-induced delays plus any delays associated with application responsiveness, thus,

$$QoE_{client\text{-}server} = f(d_{app}, d_{queue} + d_{prop} + d_{proc} + d_{trans})$$

Typical end-to-end processing of user-generated events must be processed first by the intermediate (porthole) server and then (typically) cause a transmission from the porthole server to a back-end server. Thus, the QoE associated with a porthole-based session is a function of more delay contributors:

$$QoE_{porthole} = f(d_{app}, (d_{queue} + d_{prop} + d_{proc} + d_{trans})_{client\text{-}to\text{-}porthole}, (d_{queue} + d_{prop} + d_{proc} + d_{trans})_{porthole\text{-}to\text{-}app}, d_{porthole})$$

or, more generally:

$$QoE_{porthole} = f(\sum_{k=1}^{n}(d_{queue} + d_{prop} + d_{proc} + d_{trans} + d_{porthole})_k + d_{app})$$

where there are *n* portholes involved in access high security zones from low security zones.

These additional network delays and contention for the porthole service itself can cause significant decreases in the QoE. The delay increases multiplicatively as the user is forced to traverse more "porthole hops". In modeling the system, we found that transmission and propagation delays contributed negligibly to the overall performance of the network while queuing delays at the porthole host and application delays were potentially significant. This culminated in the finding that, at times, it may be preferable for a node in one area of security to be "promoted" to a higher level of security to increase performance.

IV. Security Zone Promotion

In this section, we propose a system to facilitate moving from one security zone to another (the zones are depicted in Figure 1). The rationale for this is the performance degradation associated with accessing applications through portholes. The porthole based access is inherently slower than direct client-server access as the porthole-based access adds another layer which introduces non-zero delays. In describing the re-zoning of a node in the network, we'll make several realistic assumptions: 1) there is some triggering event that causes the node to be re-zoned; 2) the node originally obtained its IP address through DHCP; 3) the loss and re-gain of link assertion at layer two (e.g. Ethernet) will trigger a DHCP lease request; and 4) upon re-zoning, all previously established TCP/IP network connections will be terminated. This triggering event (discussed in more detail later) could be a threshold violation or a user-initiated event. One could envision an icon on a computer workstation where the "glass is broken" is escalate the user to the next security level.

A. Security Discussion

The goal of the network architecture is to ensure preservation of the BLP security model. This proposed process ensures that data from the high security zone does not enter the low security zone. By changing the designation of the node from "low" to "high" security, we are essentially taking low security classification data and placing it into higher security classification zone which is permissible under BLP. The risk associated with doing this is the potential for malware or other undesirable data or code to enter into the high security zone. Although this risk must be managed, it does not violate BLP. We also assume that it is not feasible for an unauthorized individual or node to promote another node. We have not yet developed the details of the promotion mechanism.

B. Procedure

The procedure of the system is the following: i) communications with central service to promote node to new VLAN, ii) central services communication to network infrastructure services to change VLAN or machine, iii) network infrastructure link assertion removal from node, iv) delay associated with link assertion that is sufficient to cause node to remove network stack scaffolding; v) abnormal closure of existing network connections; vi) re-assertion of link; vii) failure of DHCP renewal and request for new DHCP lease; viii) connection to application without the need for porthole use.

C. Performance

The performance of the steps listed in the procedure is highly dependent upon implementation. We experimented with components of the process on two different operating systems (Microsoft Windows 7, Linux/Ubuntu 10.04 LTS). The entire process, under ideal circumstances, took at least 2 seconds. Minimally, the following communications must occur (Table 1). These activities are all implementation specific and thus, cannot easily be quantified generally.

Table 2: Security Zone Transition Delay Contributors

Activity	**Description**
Promotion Request	Packet from client to promotion manager
Promotion Approval	Packet from promotion server to client
VLAN Re-assignment	Promotion server to network infrastructure
Link Removal	Switch de-asserts link
Link Removal delay	Sufficient to cause network on client to deactivate
Link Assert	Switch re-asserts link, negotiates speed and duplex
DHCP Renewal	Depending upon implementation, client may attempt to renew previous IP address, which will fail

DHCP Renew Failure	Depending upon implementation, client may attempt to renew previous IP address, which will fail
DHCP Lease Request	Client will request new DHCP address
DHCP Lease Response	Server will allocate new DHCP address
DHCP Lease Accept	Client will accept new DHCP address
Initiation of Applications	Client will launch applications (application/implementation dependant)

Thus, the performance of the promotion scheme must consider the one-time performance costs of the promotion activity and the on-going performance of the node's interactions with the application post-promotion, thus:

$$\mathrm{QoE}_{\mathrm{promotion}} = f(\mathrm{d}_{\mathrm{promotion}} + \sum_{k=1}^{n}(\mathrm{d}_{\mathrm{queue}} + \mathrm{d}_{\mathrm{prop}} + \mathrm{d}_{\mathrm{proc}} + \mathrm{d}_{\mathrm{trans}})_{\mathrm{k}} + \mathrm{d}_{\mathrm{app}})$$

D. Comparison

We used transactional throughput as a measure of productivity and a proxy measure for QoE. The transactional throughput is defined as the number of user interactive events that could practically occur over a specific time period. As performance declines, we expect the number of possible user transactions to decline. User events are defined as key presses and mouse clicks. This is, of course, a simplification. A single user-driven event could cause a video to start playing or a screen to render several times which result in significant traffic. However, for GUI applications that largely consist of text (such as an EHR), this is typically not the case. We measured the possible transactions over a given time.

The cumulative possible transactions at time *t* is the sum of all of the possible transactions that could have occurred in each of the proceeding discrete time units for this sessions, less the opportunity costs associated with delays, thus:

$$\mathrm{trans}_{\mathrm{c}} = \sum_{k=1}^{t} \mathrm{trans}_{\mathrm{k_t}} - \mathrm{trans}_{\mathrm{k_t}} * \mathrm{delay}, \mathrm{trans}_{\mathrm{c}} \geq 0$$

The promotion method will be superior from the perspective of more possible transaction could have occurred when,

$$\mathrm{d}_{\mathrm{promotion}} < \sum_{k=0}^{t} \left(\mathrm{d}_{\mathrm{porthole}_{\mathrm{k}}}\right).$$

This measure does not take into account user preferences or usability issues that might otherwise sway users' decision making process.

E. Experiment

We considered several scenarios to compare the performance of the porthole based access versus the performance of the promotion mechanism. Our hypothesis is "for brief forays into zones of higher security, the porthole method would be preferable." For sustained use of systems in higher security zones, the zone promotion method would be preferable. Through our experiment, we sought to verify the hypothesis and determine the value of "brief."

First, we sought to understand the network traffic associated with user-generated events. We used Wireshark 1.4.0 to capture packets related to specific porthole technology. Since "porthole technology" is not well-defined, we connected to two technologies that may be candidates for remote application portholes: a multiuser version of Microsoft Windows with Citrix and XWindows running on Redhat Linux. To test simple activities (keystrokes and mouse-clicks), we opened "notepad.exe" on the Windows machine and initiated a remotely displayed "xterm" which was tunneled through SSH to display through XWindows. We then counted how many packets were generated by each activity. Individual keystrokes (which included transmitting the keypress event and the subsequent echoing back to the remote terminal) caused, on average three packets to be transmitted. Mouse-click events tended to result in an average of twenty-two packet transmissions.

We then created a simulation in C running under Cygwin. We measured the impact of various components of delay and only considered the elements which were likely to contribute significantly and vary between the different the porthole solution and the zone promotion solution. In order to keep the simulation manageable, we made several assumptions, specifically: the inter-node distances were kept constant at 1000m, a layer three network diameter of 5, no queuing delays in the network (but varying queuing delays on the portholes and server nodes), and the networks operate at a constant 100 Mbits/sec. We assume that the TCP sessions are already established and thus no handshake is required.

F. Results

The performance reduction for the porthole based users was highly dependent upon the number of simultaneous nodes contending for access to the portal (and thus, the network queue on the server) and the amount of delay induced by the porthole system itself (see Figure 2). We varied promotion delay, and node processing delay.

G. Discussion

The performance of connecting to protected systems is quantifiably and intuitively improved for the nodes that are members of the protected zones. However, there are drawbacks that may contribute to users' decisions not to self-promote their nodes. Once part of the protected zone, access

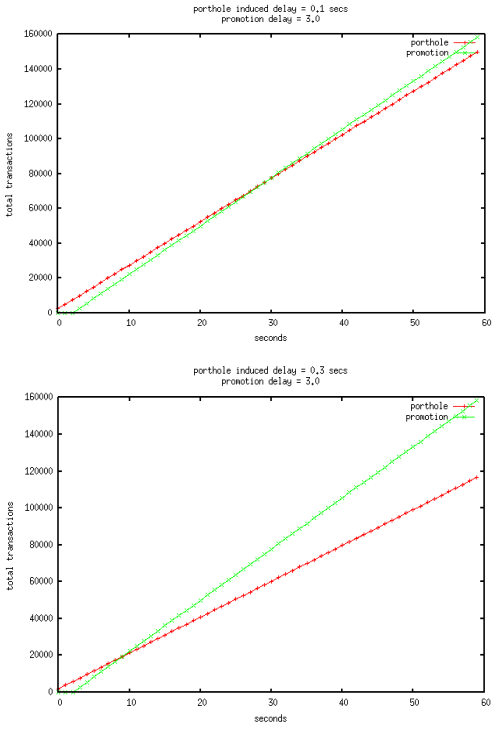

Figure 2: Varying break-even points for differing assumptions

to information assets in lower zones may be restricted or lost completely depending upon the prohibitions associated with that zone. Similarly, there are also advantages for promoting into the protected zone – for example, a user may be restricted from "copying & pasting" data – even between applications that are accessed through the porthole into the protected zone. One could also envision policy decisions that prohibit access to high security information through the porthole method. Another longer-term consideration for the user is the need to "decontaminate" the promoted node subsequent to its use in the promoted zone. There may be overall QoE costs associated with the decontamination process (time, effort) that increases their interest in the porthole based access method.

V. Future Work

In future work, we will consider the development of the secure mechanism to facilitate the node promotion process. Furthermore, we may also contemplate an automated method for zone promotion with specifically defined promotion triggers. We will also demonstrate how this security/tiered method could be utilized to ensure that hybrid computing clouds don't inappropriately offload virtual machines to inappropriate public cloud providers.

VI. Conclusion

This paper introduces two concepts: i) a VLAN based security architecture to facilitate compliance with the rules of BLP while facilitating pragmatic needs of organizations with varying classifications of data and ii) a system for promoting designated nodes to higher security zones to ensure critical access. The architecture facilitates increased protection to prevent accidental data leakage. The porthole and promotion methods both facilitate escalated access to sensitive data with different performance characteristics and access benefits and costs.

References

[1] J. Conn, "Record HIPAA settlement could portend tougher privacy enforcement." vol. 2014, 2014.

[2] J. McLean, "Reasoning about security models," Washington, DC, USA, 1987, pp. 123-31.

[3] E. J. Topol, *The creative destruction of medicine : how the digital revolution will create better health care*. New York: Basic Books.

[4] "45 CFR Part 162: HIPAA Administration Simplification: Standard Unique Health Identifier for Health Care Providers; Final Rule," D. o. H. a. H. Services, Ed. Washington, DC, 2005.

[5] "American Recovery and Reinvestment Act of 2009," in *H. R. 1* United States of America, 2009.

[6] J. W. Brady, "Securing Health Care: Assessing Factors That Affect HIPAA Security Compliance in Academic Medical Centers," in *System Sciences (HICSS), 2011 44th Hawaii International Conference on*, 2011, pp. 1-10.

[7] "Family Educational Rights and Privacy Act (FERPA)." vol. 2014 Washington, DC, 2014.

[8] C. Blackwell, "The management of online credit card data using the Payment Card Industry Data Security Standard," in *Digital Information Management, 2008. ICDIM 2008. Third International Conference on*, 2008, pp. 838-843.

[9] M. Gymrek, A. L. McGuire, D. Golan, E. Halperin, and Y. Erlich, "Identifying personal genomes by surname inference," *Science*, vol. 339, pp. 321-4, Jan 18 2013.

[10] "IEEE Standard for frame Extensions for Virtual Bridged Local Area Network (VLAN) Tagging on 802.3 Networks," *IEEE Std 802.3ac-1998*, p. i, 1998.

[11] V. Rajaravivarma, "Virtual local area network technology and applications," in *System Theory, 1997., Proceedings of the Twenty-Ninth Southeastern Symposium on*, 1997, pp. 49-52.

[12] S. A. J. Alabady, "Design and implementation of a network security model using static VLAN and AAA server," Damascus, Syria, 2008.

[13] F. Weihong and L. Aixia, "VLAN Technology Application Research based on Campus Network Security," *Applied Mechanics and Materials*, vol. 220-223, pp. 2945-8.

[14] L. Jiajia and L. Wuwen, "Security analysis of VLAN-based Virtual Desktop Infrastructure," Piscataway, NJ, USA, pp. 301-4.

[15] V. Gainer, K. Hackett, M. Mendis, R. Kuttan, W. Pan, L. C. Phillips, H. C. Chueh, and S. Murphy, "Using the i2b2 hive for clinical discovery: an example," *AMIA Annu Symp Proc*, p. 959, 2007.

[16] M. A. Siller and J. Woods, "QoE in multimedia services transmission," Orlando, FL, USA, 2003, pp. 74-6.

[17] A. Perkis, S. Munkeby, and O. I. Hillestad, "A model for measuring Quality of Experience," in *Signal Processing Symposium, 2006. NORSIG 2006. Proceedings of the 7th Nordic*, 2006, pp. 198-201.

[18] P. Casas, M. Seufert, S. Egger, and R. Schatz, "Quality of experience in remote virtual desktop services," in *Integrated Network Management (IM 2013), 2013 IFIP/IEEE International Symposium on*, 2013, pp. 1352-1357.

[19] R. Rauscher, "Cloud Computing Considerations for Biomedical Applications," in *Healthcare Informatics, Imaging and Systems Biology (HISB), 2012 IEEE Second International Conference on*, 2012, pp. 142-142.

[20] J. F. Kurose and K. W. Ross, *Computer Networking : a Top-down Approach*, 4th ed. Boston: Pearson/Addison Wesley, 2008.

[21] J. R. Dabrowski and E. V. Munson, "Is 100 Milliseconds Too Fast?," in *CHI '01 Extended Abstracts on Human Factors in Computing Systems* Seattle, Washington: ACM, 2001.