The e-Logistics of Securing Distributed Medical Data

Andrew M. Snyder
Department of Computer Science
University of Virginia
Charlottesville, VA 22904 USA

Alfred C. Weaver
Department of Computer Science
University of Virginia
Charlottesville, VA 22904 USA

Abstract - **HIPAA mandates that all healthcare data accessed or transmitted over open communications systems such as the Internet be encrypted. While encrypting a digital x-ray before storage and decrypting it before viewing will not affect a hospital's workflow, the workflow consequences of encryption on the large medical images (a single 500-slice MRI produces a 68 MB file) typically in use by a radiology department were feared but unknown. We conducted performance studies of four candidate encryption algorithms operating in a .NET environment, and then used those results in a workflow model of our hospital's radiology department to predict the impact of adding encryption to the workflow scenario.**

I. HIPAA

A. Privacy and Security

In 1996, the U.S. Congress signed into law the Health Insurance Portability and Accountability Act [1] to initiate the process of healthcare reform. HIPAA's most ambitious requirements are embodied in the so-called Privacy and Security Rules. Effective April 14, 2003, for most covered entities (and April 14, 2004, for everyone), the Privacy Rule seeks to ensure patient privacy by regulating how doctors, hospitals, healthcare plans, insurance companies, and other covered entities collect, manage, store, disclose, and utilize a patient's medical information.

The Security Rule standards, effective April 21, 2005, cover administrative procedures, physical safeguards, technical security services, and technical security mechanisms. The security services cover access control, audit control, authorization control, data authentication, and entity authentication. The security mechanisms guard against unauthorized access to data by requiring integrity controls and message authentication, access controls and/or encryption, and, if medical data is transmitted over a network (which is increasingly common), by requiring encryption, alarm reporting, audit trails, entity authentication, and event reporting.

B. Unintended Consequences

As laudable as these overall goals may be, it is their implementation that could subject the healthcare community to the law of unintended consequences. For example, the University of Virginia Medical Center is a "covered entity" under the HIPAA definitions and thus is obligated to comply with its privacy and security regulations. This in turn implies that the hospital's medical records, which are routinely exchanged over computer networks, are subject to the audit control and encryption requirements mandated for data security.

Our hospital's electronic patient record includes all diagnostic imagery acquired by the Department of Radiology, which conducts over 380,000 examinations and generates around 9 TB of data annually. While encrypting and decrypting a few digitized x-rays will probably cause no workflow problems, no one has investigated the potential workflow disruption that might result from having to first decrypt the 500 to 1000 separate images that comprise a modern computerized tomography (CT) or magnetic resonance (MR) examination. One goal of this project is to conduct that investigation and to then determine which of the allowable encryption methods are preferable, what the performance cost of encryption will be for the computers that implement them, and the resulting impact on the hospital's patient workflow.

II. ENCRYPTION

According to HIPAA, encryption is optional when dealing with a closed network. However, if information is going to be passed over an open network such as the Internet, then encryption must be utilized. There are hundreds of possible encryption methods that could be used to secure medical data, but four are of particular importance. The Data Encryption Standard [2] is a candidate because of its pioneer status; it was long the standard by which all commercial encryptions were accomplished. The algorithm was implemented in hardware for even faster encryption and decryption. Triple-DES [3] is a candidate because it is the industry follow-on to DES. It is provably more secure, but two to three times slower than DES. The Advanced Encryption Standard [4] is the newest approved encryption standard. It operates with three different key sizes (128, 196, and 256 bits), and in some implementations runs faster than DES. RSA, the best known of the public cryptographic systems [5], has scaleable security limited only by the chosen key size, but its performance is markedly slower than the other three symmetric key algorithms. Each of these algorithms offers something different, but our emphasis will

be weighing security versus speed and its impact on workflow in a healthcare environment.

A. Public Key vs. Symmetric Key

Our first observation was the disparity between the RSA public key encryption and the other three symmetric key schemes. RSA with 1024-bit keys decrypts at speeds more than 100 times slower than the other three methods; RSA with 512-bit keys is more than 40 times slower. Observed delays were 40 and 120 seconds for decrypting a 4 MB file with 512- and 1024-bit keys respectively on a 3 GHz Pentium 4 computer; these times are totally unacceptable, and thus RSA must be discarded as a HIPAA encryption candidate.

B. Symmetric Key Results

When comparing the three symmetric key algorithms, the encoding and decoding characteristics are quite similar. Fig. 1 shows how these schemes compare for encryption and decryption. It can be seen that for all files DES performs fastest and AES-256 performs slowest, although the differences were not dramatic. The algorithm that attains the highest overall throughput is DES, which averages 8.10 MB/s.

Table 1 shows the throughput of all the algorithms, sorted from highest to lowest. Each throughput measurement was averaged over the 400 tests that were preformed on each algorithm. It can be seen that DES out-performed the other algorithms in both encryption and decryption.

C. Performance Analysis

RSA showed once again that it is not a suitable encryption technique for large amounts of data. While its security is adequate for large keys, its performance is undeniably poor. RSA is best used for very small files, or for small amounts of data such as the encryption of hashes or other (symmetric) encryption keys.

The 128-bit version of 3DES performed almost identically to the 192-bit version, which is to be expected, given that they are the same algorithm. Although the 128-bit version technically has only two keys, one of the keys is duplicated in the final step of the algorithm. So while in some of the tests the 128-bit version did slightly better, in others the 192-bit version performed better, but neither was ever dominant by more than 1%.

As expected, DES performed better than 3DES but not by much. Our initial expectation was that 3DES would be about half as fast as DES; however, upon closer examination, it became apparent that the throughput on the 68 MB set of files was approximately 70% as fast as the throughput on the other files. Other than size, the only difference was in the number of files being encrypted and decrypted, which suggested that file overhead was a significant factor.

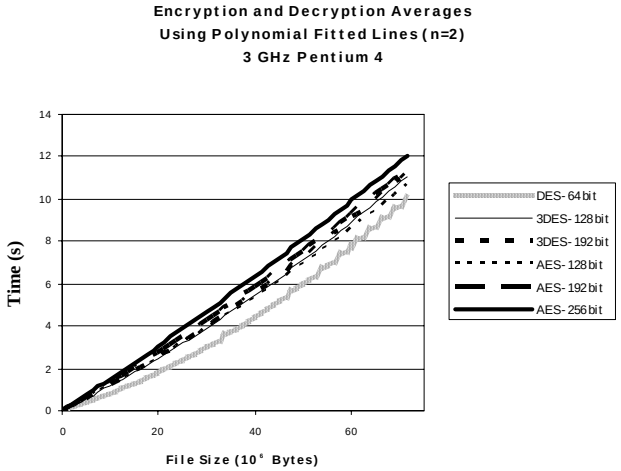

Fig. 1 Encryption and Decryption Average Times for Three Symmetric Key Algorithms

Combining the 500 separate MRI files into one 68 MB file (the conglomerate of the MRI set) produced throughput measurements along the lines of the other single file encryptions and decryptions, confirming our suspicion about file overhead.

To further separate the performance effect of encryption algorithm from system overhead, these measurements were repeated using a 600 MHz Pentium 3 machine with 512 MB RAM running Windows XP. After performing identical tests, the throughputs fell into almost the same ordering, as shown in Table 2.

Fig. 2 further confirms the performance difference between the 600 MHz and the 3 GHz machines once the 500 images (files) of the MRI test case had been combined into a single 68 MB file. On a computer where the computation is the intensive driving force, DES was almost twice as fast as 3DES.

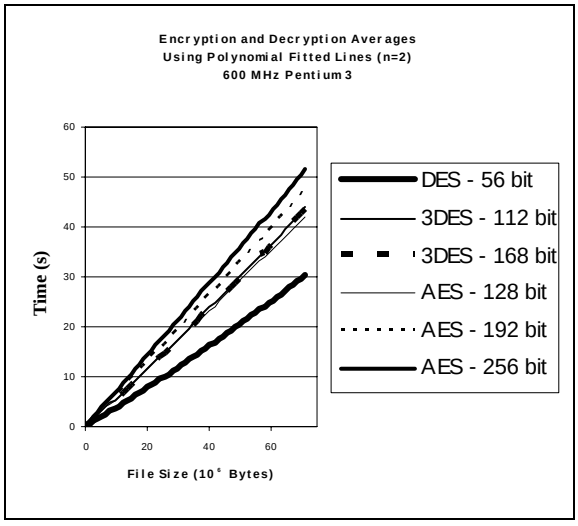

Fig. 2 Performance of DES, 3DES, and AES on 600 MHz Pentium

Table 1. Throughput of Encryption and Decryption on 3 GHz Pentium 4

Encryption Scheme	MB/s	Percent of Fastest Algorithm		Decryption	MB/s	Percent of Fastest Algorithm
DES 64-bit	8.51	100.00%		DES 64-bit	7.68	100.00%
3DES 128-bit	7.23	84.90%		AES 128-bit	6.96	90.61%
AES 128-bit	7.19	84.50%		3DES 128-bit	6.56	85.42%
3DES 192-bit	7.16	84.12%		3DES 192-bit	6.45	83.88%
AES 192-bit	6.63	77.93%		AES 192-bit	6.41	83.42%
AES 256-bit	6.24	73.36%		AES 256-bit	5.95	77.40%
RSA 512-bit	0.90	10.53%		RSA 512-bit	0.11	1.38%
RSA 1024-bit	0.62	7.34%		RSA 1024-bit	0.04	0.47%

Table 2. Throughput of Encryption and Decryption on 600 MHz Pentium 3

Encryption Scheme	MB/s	Percent of Fastest Algorithm		Decryption Scheme	MB/s	Percent of Fastest Algorithm
DES 64-bit	2.76	100.00%		DES 64-bit	2.14	100.00%
3DES 192-bit	1.81	65.38%		AES 128-bit	1.64	76.40%
3DES 128-bit	1.80	65.30%		3DES 192-bit	1.54	71.86%
AES 128-bit	1.79	64.85%		3DES 128-bit	1.53	71.57%
AES 192-bit	1.58	57.06%		AES 192-bit	1.47	68.50%
AES 256-bit	1.41	51.06%		AES 256-bit	1.37	64.06%
RSA 512-bit	0.28	10.27%		RSA 512-bit	0.03	1.39%
RSA 1024-bit	0.21	7.76%		RSA 1024-bit	0.01	0.54%

AES performed extremely well in all scenarios. The 128-bit version was even faster than 3DES overall. This makes sense given that AES was chosen by NIST because of its performance. Also, AES uses 128-bit blocks, and so it has to reference memory fewer times to get the required information (3DES uses 64-bit blocks).

This means that on any computer where memory access is the limiting factor, AES will perform better than 3DES. The 128-bit AES performed better than the 192-bit and 256-bit versions in every test, which is to be expected because it has the fewest rounds and should therefore be fastest.

D. Recommendation

Given the vastly superior protection afforded by AES with 256-bit keys, AES is the logical choice for implementing the encryption function required by HIPAA. In the workflow model that follows we assume that AES is the encryption algorithm of choice, and we utilize our own AES performance measurement of 6.81 MB/s (in the .NET environment, on a 3 GHz Pentium 4, using managed code) as its sustained encryption speed.

III. WORKFLOW IMPACT

A. Radiology Department Workflow Model

The Department of Radiology at the University of Virginia conducts over 380,000 examinations and generates approximately 9 TB of data a year. This is why the Radiology department is an excellent candidate upon which to base the workflow model.

In order to model the workflow in the Radiology department, it is important to understand how the department operates and how data flows. Fig. 3 is a dataflow diagram of the University of Virginia's Department of Radiology's PACS (Picture Archive and Communication System) that integrates with the other vital components of the department, such as the Hospital Information System (HIS), Radiology Information System (RIS), Digital Imaging and

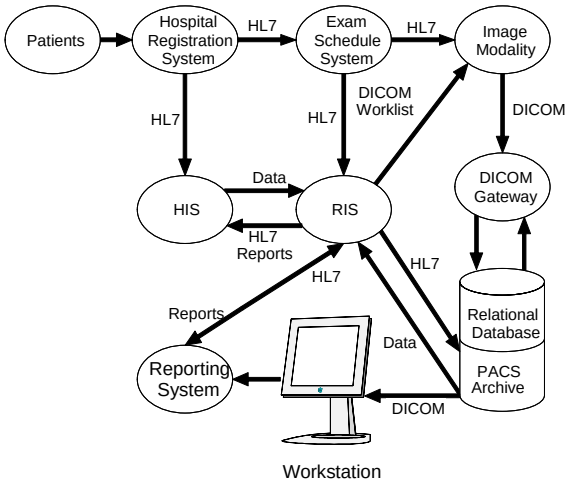

Fig. 3. Information Workflow for U.Va. Radiology

Communications in Medicine (DICOM) gateways, various image modalities, reporting systems, Health Level 7 (HL7) communications and SQL databases. This model is used by the University of Virginia to conduct throughput assessments and locate bottlenecks in its clinical procedures.

B. Workflow Architecture

In the proposed system, there are many steps that need to be accomplished from the moment a patient walks in the door until the department sends its report to the HIS. Each of these steps has associated resources that are used. When multiple steps use the same resource, it degrades the overall throughput and creates potential bottlenecks. This makes it important to create a set of steps that mimics the actual system and uses the available resources in order to find bottlenecks and the impact of the additional encryption/decryption steps. Table 3 contains 13 steps that are based upon a typical patient encounter in the radiology department.

Each of the resources is a potential bottleneck depending on which steps use them. Table 4 shows a list of the thirteen resources used in this system.

The list of resources is used in combination with the steps in the system to form a Resource Allocation Table (Table 5). It shows the resources that each step uses and is used to calculate the throughput of each of the resources.

In the resource allocation table, each step either uses resource Rx, noted as a one, or does not use resource Rx, noted as a zero. At the bottom of the table is the name of the bottleneck (Bx) associated with each resource. In the right-hand column is the time required to complete a given step, Ti. The total of all the times ($\sum$Ti) yields the average time to complete one job. In order to determine the bottleneck(s) in the system, one computes the throughput of each of the possible bottlenecks because the system can only operate as fast as its slowest bottleneck. In order to compute the bottlenecks, one solves Bx = 1 / (Ti + Tj + … + Tz) where resources i, j, …, z are used. The times that are used are the times associated with the steps that use resource Rx. For example, B13 = 1 / (T4 + T7 + T8 + T9 + T10). The unit of the result Bx is jobs per second. The resource bottleneck equations are listed below in table 6.

In order to solve each of the above equations, the average time to complete each step is required. These times were collected using a mini-PACS system at the University of Virginia Hospital without using encryption, and are listed below in Table 7. The average times with encryption are estimated based upon accessing the MRI examination with an encryption/decryption rate of 6.81 MB/s as previously measured for 256-bit AES [6].

Using the times from Table 7 and the equations from Table 6, the values in Table 8 were calculated as possible bottlenecks.

C. Workflow Results

From these values, it can be seen that the encryption is not the bottleneck when considering all the system resources, nor does it change the actual bottleneck resource. However, the encryption does lower the throughput of the bottleneck. If we were dealing with a single doctor, and he was not using encryption, the resource bottleneck would allow up to 62 patients per 24 hours. If the same doctor was using encryption, he would be able to see up to 59 patients per 24 hours. So encryption has the effect of degrading performance 5% in a system that is optimally concurrent. In the worst case scenario in which each patient must be seen and all pertinent documents completed before the next patient can be registered, the system performance will be lowered by 7%.

In order to understand the effect of encryption on patient throughput, especially when considering concurrent patients, it is important to establish bounds. In order to graph these bounds, it is important to note that Te is the time required to do the encryption when considering the life-cycle of the patient. Ts is the time spent completing the rest of the required steps for a patient. If we can assume that our system contains one pipeline and each step is independent of all other steps then we can establish the following bounds on the mean throughput rate.

In the best possible scenario, each patient is independent of each other patient, and does not have to wait for any step, which leads to an absolute upper bound of N/(Te + Ts). As N increases, this bound becomes unattainable, as there are not enough resources.

The maximum attainable throughput is achieved by considering solely the bottleneck and is not affected by the number of patients waiting in the queue. This bound is equal to 1/ Tb, and in our case Tb is equal to T6, as the patient exam clearly takes the longest to complete.

The actual upper bound throughput is dependent on the number of patients in the queue, and the availability of the

Table 3. Typical Steps in the Radiology Department

Steps	**Description**
A	Patient Registration by hospital registration system
B	Notify HIS of patient and data using HL7
C	Schedule exam and notify RIS
D	Patient data to RIS and to PACS archive
E	DICOM worklist to image modality
F	Conduct patient exam
G	Patient image data to gateway using DICOM
H	Relational data to gateway (required prior images)
I	DICOM image data from gateway to PACS archive
J	DICOM image data to workstation from PACS archive
K	Patient report generated in reporting system
L	Patient report send to RIS from reporting system
M	Patient report sent from RIS to HIS

Table 4. Resources in the System

Resource	**Description**
R_1	Hospital Registration System
R_2	HIS (Hospital Information System)
R_3	RIS (Radiology Information System)
R_4	Examination Schedule System
R_5	HL7 Communications for Text Data
R_6	DICOM Communications for Image Data
R_7	Image Modality Unit
R_8	DICOM Gateway
R_9	Relational Database
R_{10}	PACS Archive
R_{11}	Workstation
R_{12}	Reporting System
R_{13}	Encryption/Decryption Application

Table 5. Resource Allocation Table

STEP	R_1	R_2	R_3	R_4	R_5	R_6	R_7	R_8	R_9	R_{10}	R_{11}	R_{12}	R_{13}	TIME
A	1	0	0	0	0	0	0	0	0	0	0	0	0	T_1
B	1	1	0	0	1	0	0	0	0	0	0	0	0	T_2
C	0	0	1	1	1	0	0	0	0	0	0	0	0	T_3
D	0	1	1	0	1	0	0	0	0	1	0	0	1	T_4
E	0	0	1	0	0	1	1	0	0	0	0	0	0	T_5
F	0	0	0	0	0	0	1	0	0	0	0	0	0	T_6
G	0	0	0	0	0	1	1	1	0	0	0	0	1	T_7
H	0	0	0	0	0	1	0	1	1	0	0	0	1	T_8
I	0	0	0	0	0	1	0	1	0	1	0	0	1	T_9
J	0	0	0	0	0	1	0	0	0	1	1	0	1	T_{10}
K	0	0	0	0	0	0	0	0	0	0	0	1	0	T_{11}
L	0	0	1	0	1	0	0	0	0	0	0	1	0	T_{12}
M	0	1	1	0	1	0	0	0	0	0	0	0	0	T_{13}
	B_1	B_2	B_3	B_4	B_5	B_6	B_7	B_8	B_9	B_{10}	B_{11}	B_{12}	B_{13}	

Table 6. System Resource Bottlenecks

Bottleneck	Equation
B_1	$1 / (T_1 + T_2)$
B_2	$1 / (T_2 + T_4 + T_{13})$
B_3	$1 / (T_3 + T_4 + T_5 + T_{12} + T_{13})$
B_4	$1 / (T_3)$
B_5	$1 / (T_2 + T_3 + T_4 + T_{12} + T_{13})$
B_6	$1 / (T_5 + T_7 + T_8 + T_9 + T_{10})$
B_7	$1 / (T_5 + T_6 + T_7)$
B_8	$1 / (T_7 + T_8 + T_9)$
B_9	$1 / (T_8)$
B_{10}	$1 / (T_4 + T_9 + T_{10})$
B_{11}	$1 / (T_{10})$
B_{12}	$1 / (T_{11} + T_{12})$
B_{13}	$1 / (T_4 + T_7 + T_8 + T_9 + T_{10})$

Table 7. Average Times for Each Step in the System

Time	Average Time without Encryption (seconds)	Average Time with Encryption (seconds)	Short Description
T_1	900	900	Patient registration
T_2	5	5	Notify HIS of patient
T_3	30	30	Schedule exam
T_4	10	11	Patient data to RIS and PACS
T_5	10	10	Worklist to image modality
T_6	1200	1200	Conduct patient exam
T_7	180	240	Patient image data to gateway
T_8	180	240	Relational DB images to gateway
T_9	180	240	Image data from gateway to PACS
T_{10}	120	180	Image data to workstation
T_{11}	120	120	Patient report generation
T_{12}	30	30	Patient report to RIS
T_{13}	30	30	Patient report from RIS to HIS

Table 8. Bottleneck Results in Patients/Hour

Bottleneck	Equation	Without Encryption (patients/hour)	With Encryption (patients/hour)
B_1	$1 / (T_1 + T_2)$	3.98	3.98
B_2	$1 / (T_2 + T_4 + T_{13})$	79.92	78.26
B_3	$1 / (T_3 + T_4 + T_5 + T_{12} + T_{13})$	32.73	32.43
B_4	$1 / (T_3)$	120.00	120.00
B_5	$1 / (T_2 + T_3 + T_4 + T_{12} + T_{13})$	34.29	33.96
B_6	$1 / (T_5 + T_7 + T_8 + T_9 + T_{10})$	5.37	3.96
B_7	$1 / (T_5 + T_6 + T_7)$	2.59	2.48
B_8	$1 / (T_7 + T_8 + T_9)$	6.67	5.00
B_9	$1 / (T_8)$	20.00	15.00
B_{10}	$1 / (T_4 + T_9 + T_{10})$	11.61	8.35
B_{11}	$1 / (T_{10})$	30.00	20.00
B_{12}	$1 / (T_{11} + T_{12})$	24.00	24.00
B_{13}	$1 / (T_4 + T_7 + T_8 + T_9 + T_{10})$	N/A	3.95

bottleneck resource. This throughput is equal to N/(Te + Ts + (N-1)*T6). This models the effect that each patient has to wait some amount of time for all the previous patients to complete. This bound is always less than or equal to the minimum of the absolute and attainable bound. As N approaches infinity, this bound will approach 1/Tb.

The lower bound assumes that there is no pipeline but rather a sequential execution of the steps and each patient must wait for the previous patient to complete all steps before entering into the system. This lower bound is simply 1/(Te+Ts), and is unaffected by the number of patients waiting in the queue.

If one uses the preceding bounds to graph the mean throughput, it can be seen that the encryption does not affect the attainable throughput (1/Tb) since the bottleneck is unaffected by the encryption. However, it does affect the absolute upper bound, the actual upper bound convergence speed, and the lower bound. See fig. 4.

Since T6 is identical for the system with and without encryption, one can plot the actual upper bounds and lower bounds for both systems in order to see the performance difference given N patients in the system. See fig. 5. Note that the absolute upper bound will be different as well.

The actual operating mean throughput rate is somewhere between the upper and lower bounds. This is why it is important to have a solid system implementation. The graphs show that increasing the concurrency of patient encounters reduces the performance impact of using or not using encryption. However, if the system is fairly sequential, the performance penalty for encryption will be steady. These results suggest that encryption will not adversely affect the healthcare industry if reliable pipelined implementations are in place.

This is a welcome result for our radiology department. Prior to this study, the impact of HIPAA was feared but unknown. Based upon our modeling of the University of Virginia Radiology Department, we can now predict a 5-7% decrease in patient throughput. While no decrease is desirable, these results suggest that no panic is warranted. The projected decrease in patient throughput attributable to encryption is small, and can be overcome by increased efficiency in other steps.

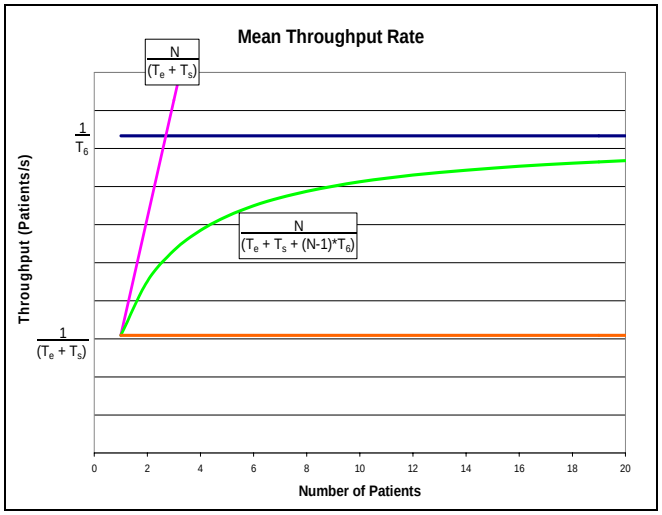

Fig. 4. Throughput of System with Encryption

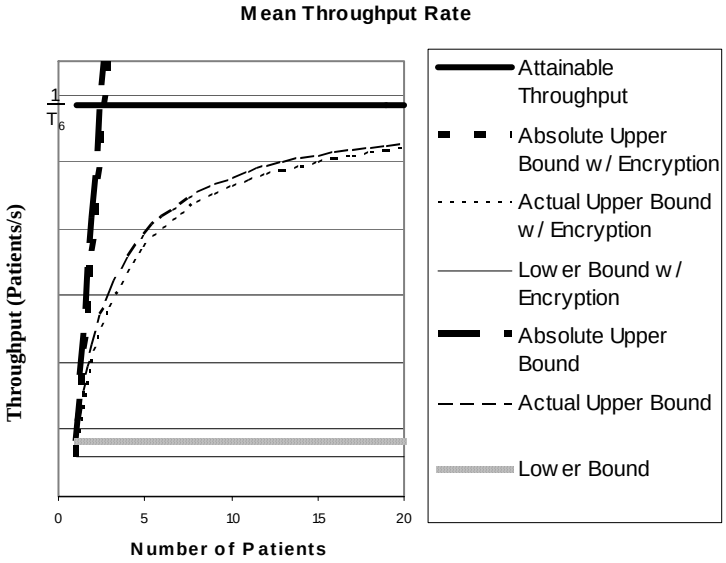

Fig. 5. Comparison between Throughput with and without Encryption

IV. CONCLUSIONS

We measured the performance of four encryption algorithms (DES, 3DES, AES, and RSA) in the .NET environment of our prototype distributed healthcare system. Due to its vastly superior security, we recommend the use of AES with 256-bit keys and recorded its sustained throughput (using a 3 GHz Pentium 4) at 6.81 MB/s. We utilized our hospital's standard workflow model for a typical patient encounter in the radiology department and modified it to account for the use of encryption in all data storage and transmission. Solving for potential bottlenecks, we showed that the likely impact would be a patient throughput reduction on the order of 5-7%. We conclude that the type of data security mandated by HIPAA will not dramatically affect the radiology department's workflow.

V. ACKNOWLEDGEMENTS

The Internet Commerce Group gratefully acknowledges the support of Microsoft Corporation and its University Research Program in conducting this study. Mr. David Ladd is our liaison at Microsoft. We acknowledge the continuous contributions over many years of Dr. Samuel J. Dwyer III in the university's Department of Radiology who supplied the workflow model and numerous constructive comments.

VI. REFERENCES

[1] Health Care Portability and Accountability Act, Public Law 104-191, http://aspe.hhs.gov/admnsimp/pl104191 .htmT

[2] "Data Encryption Standard (DES)", FIPS 46-3, http://csrc.nist.gov/publications/fips/fips46-3/fips46-3.pdf

[3] "Triple-DES (3DES)", http://csrc.nist.gov/cryptval. des.htm

[4] "Advanced Encryption Standard," FIPS Publication 197, November 26, 2001, http://csrc.nist.gov/ publications/ fips/fips197/fips-197.pdf

[5] RSA Security Inc., http://www.rsasecurity.com/

[6] Andrew M. Snyder, "Performance Measurement and Workflow Impact of Securing Medical Data using HIPAA-mandated Encryption in a .NET Environment," master's thesis, Department of Computer Science, University of Virginia, August 2003.