JACOBS PUBLISHERS

OPEN ACCESS

Jacobs Journal of Radiation Oncology

Research article

Database Verification Using Cryptographic Secure Hash Algorithm following Eclipse/Aria Version Upgrade and Database Migration

Zhigang Xu[1*], Jamie Todd Baker[1] and Samuel Ryu[1]

[1]*Department of Radiation Oncology, Stony Brook Medicine, Stony Brook, New York, United States*

**Corresponding author: Dr. Zhigang Xu, Department of Radiation Oncology Stony Brook Medicine 100 Nicolls Rd Stony Brook, NY 11790, Tel: 631-444-3617; Email: Zhigang.xu@stonybrookmedicine.edu*

Received: 09-13-2016

Accepted: 10-25-2016

Published: 10-26-2016

Copyright: © 2016 Zhigang Xu

Abstract

Software upgrades of Aria Electronic Medical Record (EMR) and Eclipse Treatment Planning System (TPS) require that all data be migrated from one version of the database to another. It is necessary to verify that the data is correctly migrated to assure patient safety. Traditionally Quality Assurance (QA) checks of a patient's radiation therapy plan and dose information involve printing out treatment parameters of each patient from the Eclipse TPS and Aria EMR system and visually checking the information for one-to-one correspondence. These QA checks performed to detect errors that are introduced during the migration can be time-consuming and inadequate especially in a paperless environment. In this work, we developed an automatic verification method to make sure that patient data has been correctly migrated. The auto verification method utilizes Eclipse Scripting Application Programming Interface (ESAPI), Cryptographic Hash Algorithm SHA-256 and Microsoft Excel. This method was used as part of our software upgrade and database migration from Varian's Aria/Eclipse 11 to 13. Plan parameters and dose records were verified for 73 active patients in 60 minutes. Compared to manual line-by-line checking, this direct comparison resulted in time savings and reduced potential human errors. The scripts can be integrated into Eclipse or can be run as a stand-alone executable program for a more automated process.

Keywords: Database migration; Eclipse Scripting Application Programming Interface; Electronic Medical Record

Introduction

The Aria EMR and Eclipse TPS use a large database running on a networked server to provide treatment plan information for delivery at the linear accelerators and to record the treatment history and other data. When the database is upgraded to a newer version, a database migration is required to map information from the older version to the new one. Maintaining the integrity of the database is of the utmost importance to assure accurate and safe treatment after the migration. The incorrect data or the incorrect operation, use, installation, and maintenance of the EMR can have disastrous consequences, potentially delivering fatal doses of radiation in a very short time [1-3]. Traditionally QA checks after database migration have been done by visually checking the information on a one-to-one correspondence. It is a time-consuming process and sometimes it is inadequate in detecting errors that are introduced during the migration. Hadley et al developed software to compare treatment plans between different versions of the EMR by translating the same plan into an XML schema. A plan comparison module takes the two XML schemas as input and reports any differences in parameters between the two versions of the same plan by applying a schema mapping [4]. The purpose of this work is to develop an automatic verification method by direct comparison of two hash values using ESAPI, SHA-256, and Excel. The method developed here was used in conjunction with an EMR and TPS upgrade and migration from Varian Medical Systems Aria 11 to Aria 13.

Materials and Methods

Eclipse and ESAPI

Eclipse is a treatment planning system by Varian (Varian Medical Systems, Inc., Palo Alto, CA). Eclipse is used to plan external beam irradiation with a photon, electron, and proton beams,

as well as for internal irradiation (brachytherapy) treatments. Eclipse is part of Varian's integrated oncology environment. The Eclipse Scripting Application Programming Interface (ESAPI) is written in C# and uses a Microsoft .NET class library that allows us to access the treatment planning data of Eclipse. Furthermore, it allows us to create scripts that can leverage the functionality of Eclipse, and lets us retrieve plan, image, dose, structure, and DVH information from the Varian System database. In the current study, scripts were created using C# with Visual Studio Express (Microsoft Corporation, Redmond, WA), and data was read from the External Beam Planning workspace.

Cryptographic Hash Function

Hash functions are mathematical computations that take in a relatively arbitrary amount of data as input and produce an output of fixed size. The output is always the same when given the same input. The inputs to a hash function are typically called messages, and the outputs are often referred to as hash values (or message digests). Nearly any piece of data can be defined as a message, including character strings and binary files. In the current study, the information retrieved by ESAPI was used as our input message. Cryptographic hash functions, which are much more powerful than regular hash functions, have the property that it is very difficult to find two different input messages that produce the same hash value. Two distinct messages that result in the same hash values are called collisions. Since different messages almost always produce different hash values, one can conclude that if a hash value of a file changes, then the file itself has changed. Since collisions are extremely unlikely to occur, if the new hash value matches the original hash value, it is extremely likely that the database has not been altered. Therefore, we see that the properties of cryptographic hash functions can be used to verify that the database has not been altered; one can quickly determine file integrity. Notice though that we cannot determine specifically what contents of the message have changed, only that something in the message has changed.

Secure Hash Algorithm 2 (SHA-2) is a set of Cryptographic Hash function designed by the National Security Agency (NSA) [5]. The SHA-2 family consists of six hash functions with hash values that are 224, 256, 384 or 512 bits. SHA-256 and SHA-512 are novel hash functions computed with 32-bit and 64-bit words, respectively. They use different shift amounts and additive constants, but their structures are otherwise virtually identical, differing only in the number of rounds. SHA-256 was used in this work and is available as part of windows cryptography library.

System design

Figure 1 depicts the verification process that was performed before and after Eclipse/Aria database upgrade. As shown in Figure 1(a), ESAPI was used to create a script to retrieve plan parameters from the Eclipse database for each patient on treatment. SHA-256 was used to generate a fixed length hash value (HV) from the retrieved plan information for each patient.

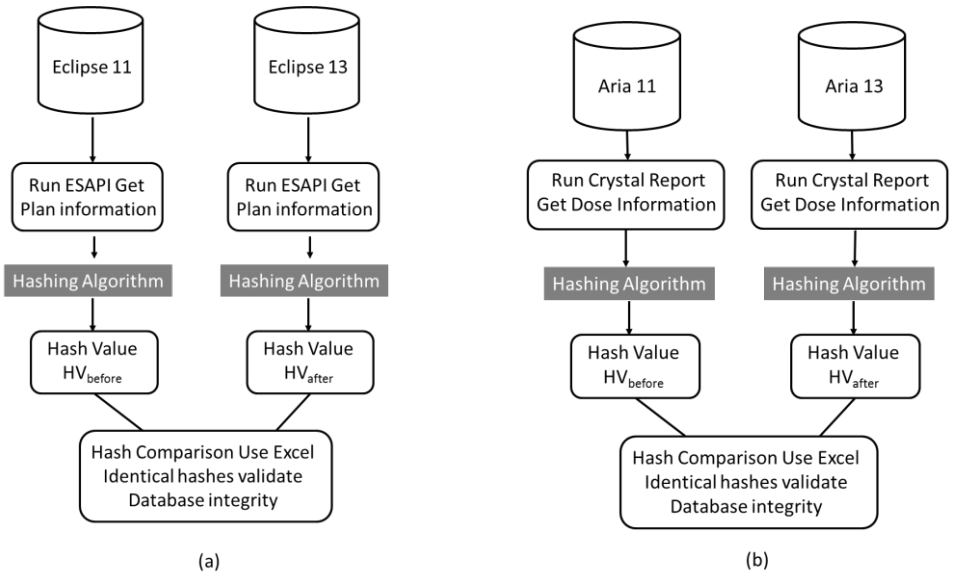

Figure 1. Diagram of verification process performed before and after Eclipse/Aria Database migration. (a) Hash values comparison process to verify treatment plan information for each patient, and (b) Hash value comparison process to verify patient dose information for all patients.

The script was run twice before and after the Eclipse database upgrades to create two hash values for each patient, HV_{before} and HV_{after}. An Excel macro was used to import the generated hash values for comparison. If HV_{before} is the same as HV_{after}, then the database is considered unchanged based on the collision-resistance property of the cryptographic hash function, that is, it is almost impossible to find two different inputs that generate the same hash value. Otherwise, we concluded that the patient data may have been altered during the upgrade process. Separately, as shown in Figure 1 (b), a crystal report was used to summarize the patient dose information for all patients on treatment from the Aria database before and after database migration. And then two hash values were generated using SHA-256 based on the crystal reports. By comparing the two hash values, the integrity of the database is verified.

Results and Discussion

This auto data verification method was implemented during the latest Aria/Eclipse V13 upgrade. Dosimetric and plan-specific information for 73 active patients were verified after the database migration. Among these patients, only one patient had a different HV. Upon investigation, the difference was found to be due to the patient name being changed to upper case during the database conversion. Varian was informed and the name was corrected accordingly. The patient dose information for 73 patients was verified by comparing two hash values generated from the crystal dose report. The two hashes were found to be identical which validated the database integrity after the migration. The time to hash 73 patients was less than half hour. The whole verification process took about 60 minutes.

Software upgrades require careful checking of database information stored in the database to assure patient safety and guarantee that the treatment delivery data are correct. The impact of incorrect plan data caused by a faulty migration could be devastating for a patient. With ESAPI, it is easy to create customized add-on software libraries that can be installed in the planning system as a pull-down menu item. Many programs and scripts have been created since the upgrade of Aria Version 11. Some examples of the programs include the ability to analyze the patient's Dose Volume Histogram (DVH), to acquire analytic metrics concerning the patient dosimetry (Dose-at-Volume, Volume-at-Dose, Homogeneity Index (HI), Conformity Index (CI), etc.), the ability to automate some treatment planning system quality assurance and validation of commissioning data, and the creation of automated patient reports.

The properties of cryptographic hash functions have many applications in the realm of computer security and data migration. Programs built on top of cryptographic hash functions have the ability to help us detect changes of EMR database after database migration. These concepts are particularly relevant in a growing cloud network where a large database running on a remote networked server communicates with clients through Citrix.

Conclusion

We developed an automatic database verification method using ESAPI, SHA-256 Cryptographic Hash Algorithm, and Microsoft Excel. Without this method, the time to verify a single patient's information was about five minutes. For 73 patients, this would result in 6 hours of work. With this method, we can verify 73 patients in less than an hour - resulting in time savings of five hours. Compared to a manual checking process, this process reduces the potential human error associated with tedious one-by-one checking and also allowed for more parameters to be checked at a more detailed level at a fraction of additional labor cost. These scripts can be integrated into Eclipse or can be run as a stand-alone executable program for a more automated process and even more time savings. This approach can easily be applied to other Treatment Planning and EMR systems.

References

1. Siochi RA, Balter P, Bloch CD, Bushe HS, Mayo CS et al. Information technology resource management in radiation oncology. J Appl Clin Med Phys. 2009, 10(4):16-35.

2. Siochi RA, Balter P, Bloch CD, Santanam L, Blodgett K et al. A rapid communication from the AAPM Task Group 201: Recommendations for the QA of external beam radiotherapy data transfer. AAPM TG 201: Quality assurance of external beam radiotherapy data transfer. J Appl Clin Med Phys. 2011, 12(1): 170-181.

3. Siochi RA, Pennington EC, Waldron TJ, Bayouth JE et al. Radiation therapy plan checks in a paperless clinic. J Appl Clin Med Phys. 2009, 10(1): 43-59.

4. Hadley SW, White D, Chen X, Moran JM, Keranen WM. Migration check tool: automatic plan verification following treatment management systems upgrade and database migration. J Appl Clin Med Phys. 2013, 14(6): 350–358.

5. Secure Hash Standard (SHS). FEDERAL INFORMATION PROCESSING STANDARDS PUBLICATION, Information Technology Laboratory, National Institute of Standards and Technology, Gaithersburg, MD 20899-8900, August 2015.