Securing the HealthHeritage Web, a Family Health History Web Site
James R Barrett, Ph.D. 1, William D. Woolfolk1, Douglas P. Wagner, Ph.D. 1, Anthony J. Wenzel 2, and William A Knaus, M.D.1

1 Department of Health Evaluation Sciences, University of Virginia, Charlottesville VA.
2 Dominion Digital Inc, Charlottesville VA.

Background
Progress in human genomics has major implications for clinical practice: much information is now starting to be known about the genetic basis of diseases. The HealthHeritage web project is a prototype web site for consumers to: (1) enter family health histories and, (2) evaluate the results with their primary care physician. Computer security has been a primary concern in the web development. While some of the security procedures for the operating system and web server use conventional techniques, other areas, such as our database server communication protocol, and data obfuscation process, are new or unusual techniques.

Methodology
We began our investigation of security by creating a security policy document for the web project. In this document, security is divided into several areas: (1) physical, (2) operating system, (3) web services and network transmission, (4) database server, (5) obfuscation of the data, and (6) audit procedures. The policies document helped to organize and formalize our procedures and also to have some guidelines in place in case security is violated, including a disaster recovery plan.

Results
Physical security
The University of Virginia Hospital offered the HealthHeritage project computer space in a secure room. This room is designed to securely house systems containing confidential hospital information.

Operating system and web server
We chose to use Microsoft Corporations’ NT operating system and their Internet Information web server (IIS) for our project. We used ColdFusion for the web to database services. We used the Microsoft TechNet knowledgebase checklist for NT server and IIS security issues to determine what features of the operating system and web server would be disabled or removed.

The web application software uses two techniques to authenticate users and validate their privileges. First, users must register on line to use the main part of the web server. A new user registers by providing a login name and user selected password plus a key identifying their primary care physician. The login name and password are subsequently used for logging into the main web site. Users that forget their password must phone in and answer questions about their recent web activities. We feel this is an appropriate balance between security and usability. This is a key security decision: we must make it easy for our consumers to use the system but protect them from curious relatives that may be able to guess passwords based upon family information (e.g., mothers’ maiden name).

We also implemented the use of programmer defined Coldfusion tags that allow the application program to determine if the current user has the right to view a given page or portion of a web page.

Database server
The database is stored on a separate SQL 7.0 server that runs no network protocols other than NETBEUI. In our environment NETBEUI packets are not transmitted past the local router.

Data obfuscation
We chose to hide user entered data among false records. The database could be particularly vulnerable at the beginning to curious relatives guessing among a few records. We initialized our database with false records. Identification of false records, along with all other personal data are stored in separate structures with very limited access.

System audits
We believe the best defense against security break-ins is human audits. Such audits were conducted during software development, and daily audits are scheduled throughout the expected life of this web site.

Conclusions
We implemented almost all measures recommended by our operating system, Web server and web to database vendors. Unimplemented measures included features vendors stated to have a low to moderate risk and we determined the implementation cost was high. Special security issues for the HealthHeritage site are to protect a user from a curious relative and from advocates of extreme privacy. We solved these issues by: (1) separating the database on a machine with limited network access, (2) authenticating each user before displaying web pages with (their) data, (3) restricting access to the entire database to two system authenticated administrators, and (4) hiding the actual data records among falsified records.