Paper 44-27

A Secure Online Data Validation & Collection System for HEDIS® Survey Data

Christopher A. Roper, National Committee for Quality Assurance, Washington, DC. USA
Yong Li, National Committee for Quality Assurance, Washington, DC. USA

ABSTRACT

The National Committee for Quality Assurance (NCQA) collects and reports performance and customer satisfaction survey data on managed-care organizations. The health plans (or their data processing vendors), submit this information to NCQA as part of the accreditation process. In the past, data was collected via 3 ½- inch floppy diskettes. One critical component of the data collection process was the validation of the data files. These data files were required to adhere to strict rules concerning file layout and data interdependencies. It was very common for a data file to be submitted to NCQA, rejected by the NCQA data validation process, and returned to the sender for correction. With thousands of files to collect and validate, the task of tracking and processing these diskettes was unwieldy. NCQA devised a system using base SAS® (with Macro language and SCL), a little ASP code and Microsoft SQL Server. This new data collection system proved to be far more efficient, timely, and less stressful for all involved.

INTRODUCTION

This paper will describe the Web-based data validation and collection model NCQA uses to capture HEDIS and related data, and report back to the health plan and vendors the results of the data validation process. The Data Collection System (DCS) will be broken down into three broad processes - Data Submission, Data Validation, and Data Reporting. Each of these categories will be further divided into subcategories to explain the business functions that the system is required to support, and the technical solutions that were used to meet those business needs.

DATA SUBMISSION

First, an explanation of the 'old' system will give an idea of the problem that needed to be fixed. NCQA had five account managers whose task was to receive the floppy disks from the health plans and ensure that the data on the diskettes was correctly entered into the NCQA data stores for later analysis and reporting by the research division and the accreditation divisions. The diskettes would arrive individually, in overnight envelopes, or in boxes. They would pile up in the cubicle of the account manager until he/she could process them. This processing included recording the time and date of receipt, running the diskette through a client-based data validation tool, and then e mailing the results of the validation back to the health plan. The time required to process one data submission fully would take at least one week, and often longer. Each account manager would be responsible for hundreds of these data submissions, and keeping track of where each submission was at any point in the process was extraordinarily difficult. Given that the average recidivism rate was almost three, each data submission would need to be processed three times before it passed the data validations and could be accepted. This meant the process was taking far longer to complete than was necessary, and job of coordinating, controlling, and managing the data submission was onerous. Obviously improvements had to be made.

DATA SUBMISSION BUSINESS NEEDS

The overburdened account managers were very clear in their expectations for a new Data Collection System. This system would eliminate the need for all those floppy disks flying around from the health plans to NCQA and back again. In fact, they didn't want to see any data until it had passed all of the data quality validation checks. Furthermore, they didn't want to keep track of which health plans had submitted data files and which hadn't, and of those who had which had completed the process and which were in process. In short, they only wanted to talk to the health plans about the validation rules that the data was subjected to, in order to help the health plans understand the specifications and requirements for submitting the data. The account managers wanted the 'computer' to handle everything else.

The data submission function was defined as having four main requirements. First, the solution must be deployable to hundreds of external users with unknown computer configurations. Second, the data files must be transmitted to NCQA using a secure transmission method. Third, each health plan would be able to submit a data file at any time day or night. And finally, no one from NCQA would need to physically receive the data file for it to be accepted.

DATA SUBMISSION SOLUTION

The most practicable solution for the data submission function was by necessity a web based solution. The ubiquity of the web made it the obvious choice for deploying any solution to the target audience. Security would prove to be a more

challenging requirement to meet. With the 'always on' requirement competing for the very limited Information Technology resources at NCQA, it became obvious that the web services would be hosted by a third party. This meant that our web server (www.ncqa.org) would not physically reside on NCQA premises - making security that much more difficult to implement. Furthermore, the compute processes required for validation were not to be performed on the web server. Therefore, another secure file transfer would be required between the web server and the compute server. This all had to happen at a remote site, 24 hours a day, 7 days a week, with minimal intervention by NCQA support staff. It became clear that the security systems would need to be well planned and implemented.

DATA SUBMISSION SECURITY

NCQA receives patient level detail data in much of the data files submitted - so ensuring security of the data submission process is an absolute necessity. The security systems would have to provide for two main functions - Authentication and Communications Integrity (otherwise known as Encryption).

AUTHENTICATION

For this paper, authentication is defined as two-way verification. The first verification is that the server (www.ncqa.org) is genuine, so that NCQA customers can be sure they are communicating with NCQA. This was accomplished by installing a VeriSign® Server ID on the web server. VeriSign® Server ID is a commercial third party product widely recognized as a proven, superior tool for internet server authentication using a digital server certificate. Web browsers automatically check that the NCQA web server's certificate and public ID are valid. In this way the health plans are assured of the authenticity of the NCQA web site. This satisfies the first requirement - authenticating the NCQA web site.

The second authentication requirement is to validate the identity of the health plan. With hundreds of organizations submitting data files in any one of a dozen or more layouts, understanding who is trying to submit, and what they are trying to submit is crucial to successfully receiving the data file. For user authentication, there was determined to be three levels of authentication required:

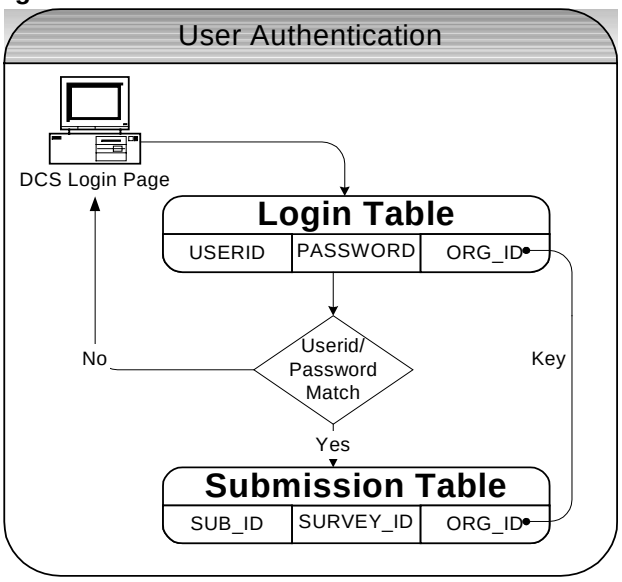
1. System Authority: Verify that the user is allowed to access the Data Collection System.
2. File System Authority: Verify that the user is allowed to view the contents of a folder or file.
3. Document Authority: Verify that the user is allowed to view the contents of a page or portion of a page.

For the data submission portion of the DCS, only the first of these is relevant. The other two will be further explained in the DCS reporting section later in this paper.

System Authority enables NCQA to control the data submission process. This is the broadest of the security levels, and defines who is allowed to send data to NCQA and what they are allowed to send. Essentially, for each type of survey for which data is submitted, there is a unique file layout. Additionally, different file formats are used based on the survey type - some of the file formats are ASCII (i.e. CSV, TXT) and some are proprietary (i.e. XLS, ZIP). All this variability required that the data submission process have a simple and reliable method for identifying uploaded files - and who should be uploading them.

NCQA uses a database access control model to implement the System Authority. Designed for simplicity, the NCQA model uses two lookup tables - Login and Submissions. The Login lookup table held the Userid, Password, and Organization ID columns. The Submissions table held the Organization ID, Submission ID and Survey ID. A Login ASP page was written to present a form to the user to enter their pre-assigned userid and password. This information was then compared to the Login table, and if a match were found, the DCS main page would be displayed. Invalid logins were returned to the Login page.

Figure 1



Once a valid user was identified, NCQA knew what data submission files that user might send. The problem became how to identify which survey a data file was submitted for. Implementing a naming convention for the data files the health plans were

submitting to NCQA solved this. The first three characters of the file name identified the survey methodology for data file, the last four characters of the file name identified the NCQA specified submission ID, which identified certain characteristics of the survey data. Finally, the file extension identified the file format that the data was submitted under. Using the Submission lookup table and some ASP code running in the background, the data files could thus be verified as the user attempted to upload them to the NCQA web server. Now that the identity of the files and senders has been confirmed, the communication channels must be made secure.

COMMUNICATION INTEGRITY (ENCRYPTION)

The challenge for ensuring communication integrity is two-fold. The first is privacy. The communications between NCQA and the healthcare organizations can not be intercepted or subject to eavesdropping by a third party. The second challenge is data corruption - either intention or accidental. Communications must not be altered en route without NCQA's knowledge. Secure Sockets Layer (SSL) encryption provides the capabilities to meet these challenges. NCQA uses an SSL server certificate (or digital key) created using VeriSign's Global Server ID®. This enables 128-bit SSL encryption - the highest level of encryption commercially available. Using this SSL encryption guarantees privacy by establishing a secure channel of communication between the NCQA web server and the healthcare organization submitting the data files. All of the data transmitted through this channel is encrypted so that even if the communication is intercepted, there is no intelligible information to be had - it just looks like gibberish. Additionally, SSL encryption is encapsulated. Encapsulation ensures that the data in the communication is not altered after it leaves the sender. Therefore, with the SSL encryption enabled on the web server, healthcare organizations can safely submit sensitive data to the NCQA web server.

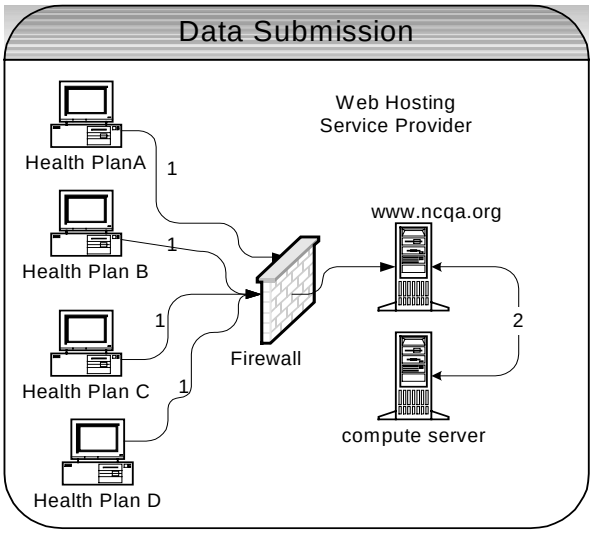
CONSTANT AVAILABLITY AND UNATTENDED RELIABILITY

The final two keys to making the data submission process successful would be to make it constantly available and automated. The healthcare organizations would need to upload data files virtually 24 hours/day as they are scattered across numerous time zones. Automation of the process would eliminate the need for NCQA staff to physically interact with the process in order to complete the data submission. The constant availability requirement was met through moving the NCQA web server to a web hosting service that was better suited to meeting the demands of a 24 hour service requirement.

Automation of the data submission process would enable no interaction or intervention from NCQA staff for a data submission to be received. One of the biggest stumbling blocks to reliability in any web site is bandwidth and network stability. In any web-based process, communications between the web server and the client (browser) can break, and often do. For most web sites, this is resolved by having the browser refreshed, and all is well. However, in a data submission process, a break in the communication process can leave the whole data submission process is limbo. Compounding this is the fact that very little of the communication infrastructure of the web is under the control of any one entity - much less anyone involved with NCQA.

There are two places in the process where a communication break can occur. First, as the data submission is being sent to the NCQA web server, and second when the web server sends the data submission file to the compute server for validation and other processing. The first vulnerability is resolved by the use of SSL channels to transmit the data as discussed earlier in this paper. The second vulnerability is less obvious and behind the scenes from the user point of view, but actually is far more troublesome. From a process standpoint, the communications between the healthcare organizations and the NCQA web server is a multi-link architecture, whereas the link between the NCQA web server and the compute server is a single link. If this single link fails, there is no alternate route of communication ready to step in. Thus it becomes obvious that the most vulnerable link is the one between the web server and the compute server. While this vulnerability can not be eliminated, it can be ameliorated by moving the compute server as close to the web server as possible. The most reliable solution from the compute server point of view would be to have the computer server and the web server the same physical machine. However, this would be a poor design from a web server point of view. The web server is busy enough serving pages to bog it down with compute responsibilities. The most practicable solution is to place a compute server physically near the web server and connect the two servers via a private LAN. See figure 2.

Figure 2



This solution has several advantages. The various health plans connect independently to the NCQA web server. A failure in any one of those links (noted as 1) does not affect any of the other health plans. Placing the compute server at the web host allows for a direct private LAN connection between the web server and the compute server (noted as 2). This connection communicates at LAN speeds - 100 mb/sec. Having the two servers physically located next to each other eliminates most of the hazards that can interrupt the communication link between them. Additionally, since the communications are occurring over a private LAN connection behind the firewall, the communications are secure and not open to interception or eavesdropping. Finally, the only requirement to now save a data file on the compute server is to map a drive on the web server that points to the compute server and save the file normally. With this design, NCQA is able to meet all the technical requirements of the Data Submission process.

DATA VALIDATION

The Data Validation process is the first part of the DCS that uses SAS® software to perform its functions. The purpose of the Data Validation process was to ensure that the data received from the healthcare organizations was useful for analysis and reporting purposes. In general, there are two levels of validation criteria - tier 1 and tier 2. The first tier is simplistic, and concentrates on ensuring that the data file submitted adhered to the published NCQA specifications for that particular survey type. In addition, tier 1 also validates data values fall within defined ranges. For example, it may check that the value in columns 3-5 are between 23 and 186 and don't contain any non-numeric character data. Tier 2 validations are more involved and perform cross data validations and validations of data interdependencies. For example, one of the Tier 2 validations is done to ensure that the survey protocols are performed appropriately, by calculating the percentage of completed survey data lines in the data submission file. Data Validation was another time-consuming repetitive process that was diverting far too much of the Account Managers' time. And to solve this problem, NCQA turned to SAS software for the solution.

DATA VALIDATION BUSINESS NEEDS

The Data Validation process had four primary business needs to satisfy in order to be a success. It had to be able to correctly process hundreds of data files on demand according to the published data submission requirements with little or no intervention from NCQA staff. Secondly, it had to do this in a secure, online environment that was available to the healthcare organizations 24 hours/day. Thirdly, a validated data submission would need to be saved automatically to the internal NCQA file servers for the Data Warehouse. Finally, the Account Managers needed to be able to quickly track every data submission attempt that was processed and quickly determine the outcome of the data validations for that submission.

DATA VALIDATION SOLUTION

The first challenge of the Data Validation process was to build a system that would process the data submission files 'on demand' - as they were placed into the Returns folder by the Data Submission process. The Data Validation process would also need to know which healthcare organization had sent the data file, and which survey type the file had been submitted. Identifying who sent the data file and for which survey type was solved by using a naming convention for the data files as they were submitted. The Data Submission process knew the healthcare organization by the userid, and it used this information to verify that the uploaded file followed the naming conventions published by NCQA. With this information, NCQA could ensure that the data submission file was validated using the correct series of validation macros.

DATA VALIDATION POLLING PROGRAM

The 'on demand' requirement for processing the data submission files was solved by writing a SAS polling program. In this instance, a polling program is defined as one that runs continuously, asking the

question 'Is there a data submission file to process in the Returns folder?' This polling program is the core of the Data Validation process. In addition to polling the Returns folder, this SAS program is also a driver for a series of SAS macro programs that perform the data validation checks on the data files submitted by the healthcare organizations. The polling program employs a macro language loop using the %DO %WHILE statement to perform the continuous folder polling. The value for the expression that is evaluated is read from an external file, the Loop Control File. See the code example below:

```
%macro loop;
filename loopcntl 'c:\temp\loop.txt';
%do %while(&loop);

data _null_;
  inflile loopcntl;
  input @1 var $1.;
  call symput('loop',var);
run;
filename loopcntl clear;
… (call validation macros)
%mend;
%loop;
```

If the first character in the loop control file is a number not equal to zero, then the %DO %WHILE loop expression evaluates as true and continues polling. Otherwise if the first character in the file is zero, the expression evaluates to false and the loop terminates. This enables NCQA to terminate the polling program at a defined location in the validation process. This is important for a number of reasons. The polling program is also a driver program. As the polling program runs, it reads the file name of the data submission file it is processing. Using this information, the polling program launches a series of SAS macro programs to perform the validation checks. Terminating the processing in mid stream could cause data results to be misreported or cause partial results to be published to the healthcare organization. Additionally, this would create a need for a robust rollback and recovery capability that would be difficult and time consuming to implement. The %DO %WHILE loop provides for a much simpler solution. Since the loop's expression is evaluated at the top of the loop, the validation process can be terminated after the current file completed the validations, and before the next data submission file begins data validations - simply by editing a one character text file.

DATA VALIDATION ENCRYPTION

Securing communications for the Data Validation process was relatively simple after solving these problems for the Data Submission process. First, most of the communications and processing happened on one physical machine - the compute server at the NCQA web host. There were only two channels of communication that needed to be secured. The first was the communication back to the web server once the validation process had completed. The existing private LAN connection created for moving the data submission files from the web server to the compute server was of course, bi-directional. This made it the obvious choice for secure communications from the compute server to the web server. The bigger challenge was securing communications between the compute server and internal NCQA file servers.

ENCRYPTED FTP FOR DATA VALIDATION

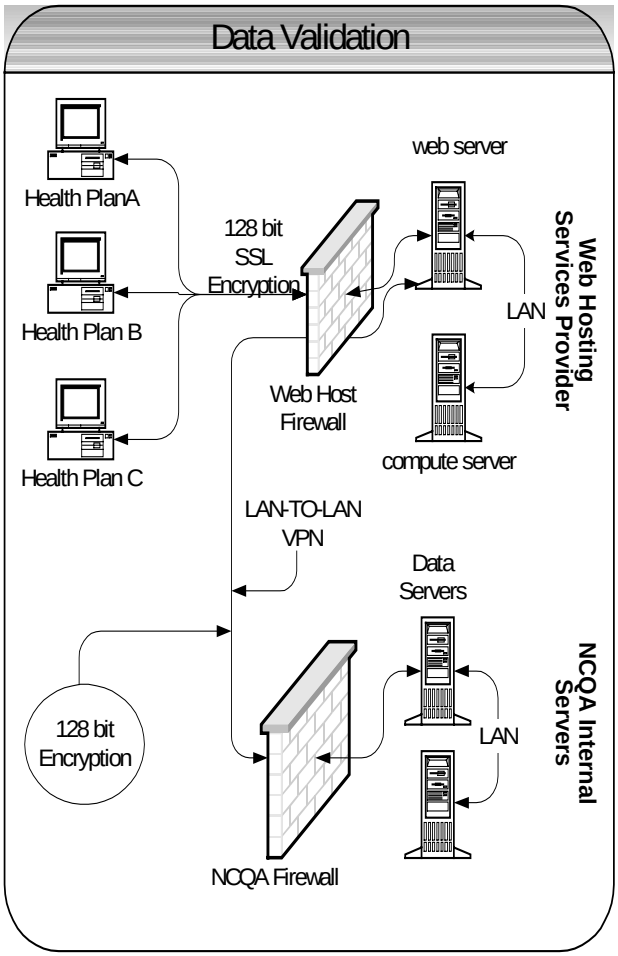
The first tests of encrypted communications between the compute server at the web hosting service and the NCQA internal file servers used FTP as the communications protocol. NCQA implemented a third party product - WS_FTP Server 3.0® from Ipswitch. This server-based tool could establish a secure FTP transfer using 128 bit SSL. NCQA tested this product and it seemed to work flawlessly. However, when the product was put into system testing using the web server and the compute server hosted at the web host service, problems arose. The web hosting company was not able to resolve port configuration problems on their firewall that would enable this secure FTP server to function properly. Without the web host's firewall ports configured as required by the WS_FTP Server 3.0 product, the FTP communications could not be encrypted, and thus this solution could not be implemented. Another solution had to be found.

PERSISTENT LAN-TO-LAN VPN

The solution that satisfied the security requirements was to establish a persistent LAN-to-LAN VPN (Virtual Private Network) connection between the private LAN connecting the web server and the compute server, and the internal LAN at NCQA. Normally, a VPN connection is used to allow a remote user access to a company's LAN. However, it is entirely possible to establish this connection between two LANs as well. With a little work from the IT Operations side of the house, this connection can be made persistent. This means that it is made permanent, so that it can be used 'on demand' by the systems on the compute server to automatically move data from the compute server at the web

hosting service to the internal NCQA file servers. To the compute server, the internal NCQA file servers appear to be on the same LAN as the web server. There is one drawback to this solution - the VPN connection is relatively slow, and thus would not be suitable for large data transfers or heavy traffic. For the Data Validation process, all the heavy traffic occurred between the web server and the compute server. The actual amount of data that needed to be transferred to the NCQA file servers was far less, and the VPN connection was sufficient for this demand. See figure 3.

Figure 3



As with the private LAN connecting the web server to the compute server, the persistent LAN-to-LAN VPN connection was bi-directional. This enabled the Account Managers at NCQA to view the processes logs created during the validation processing. These logs were invaluable in assisting the Account Managers in answering questions from the healthcare organizations about the status of their data submissions and the corresponding data validation reports.

DATA REPORTING

The Data Reporting process serves two purposes. The first is to report back to the healthcare organization the results of the validation of the data submission file. The second is to provide the healthcare organization with summary statistics and peer organization comparative statistics. For both of these purposes, SAS is used to generate the reports.

DATA REPORTING BUSINESS NEEDS

The Data Reporting process had to satisfy three business needs. First, it had to be timely - the report generation process had to produce reports and post them to the NCQA web site as soon as the validation processes were complete. Second, the security requirements were the same for the Data Reporting process as for the other of the DCS processes. Finally, the Data Reporting process had to create thousands of customized reports without intervention by NCQA staff.

DATA REPORTING SOLUTION

The Data Reporting solution is a series of macros called by the polling program immediately following the validation macros. These macros perform several tasks in the process of generating and posting the DCS reports. The first task is to report the results of the data validation checks. Once the data files passed validation, then the next task was to generate the appropriate summary statistics and peer statistics for the data file submission. The summary statistics would then be written out to a data file. This data file would be compressed and saved as a WinZip® data archive file, and posted to the NCQA web site for the appropriate healthcare organization to download. Then the Data Reporting process would generate the customized validation reports as an ASP page with an imbedded HTML frameset. Following this, the next task is to post these reports onto the NCQA web site under the appropriate directories for the healthcare organizations to access. Finally, an SQL Server® status table would be updated to reflect the results of the Data Reporting process. All these tasks are carried out by the SAS polling program. Overhanging all these tasks is the continuing need for ensuring the security of the reports - making sure that only the proper users could view the reports.

DATA REPORTING TASKS

The Data Reporting process is largely performed using Data _Null_ processing. This made the creation of the HTML framesets challenging as the

default frameset created for Data _Null_ reporting by the SAS ODS was not satisfactory for NCQA purposes. The Body reports of the frameset were produced satisfactorily by the SAS ODS system. Printing the reports to a file using the Adobe Acrobat PDFWriter® printer driver generated the PDF reports. The straight HTML reports were created with a Data _Null_ step. Other reports were created in ZIP and XLS file formats by using system calls and DDE to the appropriate application software. The problem was the menu portion of the frameset. NCQA wanted more control on the layout of the left side menu page. NCQA's solution to this problem was to design templates for these menu pages. Not ODS templates, but rather ordinary HTML frameset templates. At key places in these templates tokens were placed that would be replaced by resolved SAS macro variable values as the reports were generated. For example, the following code is from a template:

```
<A HREF="##SUBMISSID_1.html#DCS_2"
TARGET="body">Main Reports</a>
```

In this example, the tokens are ##SUBMISSID would be replaced by a value from a SAS macro variable. The following SAS code shows how this is done:

```
filename template 'c:\template.html';
filename report 'c:\user_report.html';

data _null_;
 file   REPORT   lrecl=300;
 infile TEMPLATE lrecl=300 truncover;

 input linetxt $300.;

/****************************************/
/* Search for the token ##SUBMISSID     */
/* and replace with the value of the    */
/* macro variable MAC_SUBID             */
/****************************************/
 if indexc(linetxt,'##SUBMISSID') then
    linetxt =
tranwrd(linetxt,'##SUBMISSID',"&MAC_SUBID");

/****************************************/
/* Write the new line out to the REPORT */
/* file for the user report             */
/****************************************/
put linetxt;
run;
```

This enabled NCQA to produce a frameset consistent with the design requirements of the DCS and retain all the processing within the SAS System.

DATA REPORTING SECURITY

The Data Reporting process uses the same communication channels as the Data Validation process, and thus those communications were encrypted and secure. However, there where additional security requirements beyond communication. The reports posted on the NCQA web site were to be viewed by healthcare organization that submitted the data file - and no one else. NCQA used a three tiered file security system to ensure that only authorized users accessed the system, and that they only viewed the documents they were allowed to view. The three levels of security are:

1. System Authority: Verify that the user is allowed to access the Data Collection System.
2. File System Authority: Verify that the user is allowed to view the contents of a folder or file.
3. Document Authority: Verify that the user is allowed to view the contents of a page or portion of a page.

As explained in the Data Submission section of this paper, System Authority is the broadest of the security levels. An ASP login page that allows the user to enter an userid and password provided by NCQA enables this security. With this information, NCQA is then able to determine where the user is allowed to navigate within the NCQA web site, and what reports they are allowed to view.

FILE SYSTEM SECURITY

NCQA had well over 500 individual users that would be accessing the Data Validation reports. Each user would access on average 12 separate reports. Each of these reports were HTML framesets consisting of up to 10 separate ASP pages. When a submitted data file failed validation, the healthcare organization would correct the errors and resubmit - this resulted in a new set of reports. In the end, there were well over 100,000 separate files that needed to be secured. The challenge was to design a security system that would ensure that the users could view any and all of their reports, without being able to view the reports for a different user. NCQA decided to use NT File System security to achieve this. By placing all the reports for a particular user under a single, user-specific directory, NT File System permissions could be used to prevent unauthorized users from viewing those reports. Therefore, the solution was to create 500+ folders (one for each of the users), and assign to them read access permissions based on the userid and password already provided to the users.

Now the challenge became convincing the network administrator to create the 500+ users, and assign them the proper access permissions on the web

server to the corresponding 500+ directories. Needless to say, this was not a well-received request. In order to maintain a good working relationship with the network operations staff, an automated method for completing this task needed to be found. Fortunately, there was a simple one to be had! This entailed creating three simple (but long), DOS scripts. The first script used the command MKDIR to make the directories on the web server. The second script used the NET USER command to create the users on the web server. Finally, the third script used the command XCACLS to assign the access permissions to the folders, thus ensuring that only the authorized user could view the contents of the Data Validation Reports folder for a particular healthcare organization. Using these three DOS scripts enabled NCQA to employ NT File System security on the directories containing the Data Validation Reports. Therefore, the first time a user tried to view the contents of one of these directories, the user was prompted for a userid and password. Plus, as long as the user kept the browser open, the server kept the login session active.

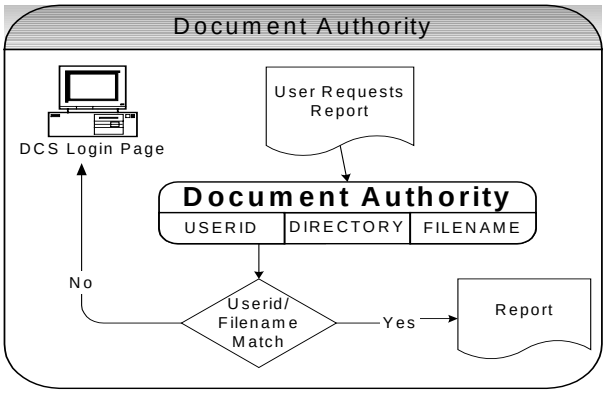
DOCUMENT SECURITY

File System security met most of the requirements for ensuring that the proper people were allowed to see their reports, and that no unauthorized people were able to view any reports. However, there were cases where a different level of access was required. Occasionally, a healthcare organization would contract the data submission of some of their data to a third party. In these cases, both the healthcare organization and the vendor would need access to the reports. One solution would be to create two copies of the reports, and put a copy into each of the users respective folders. This idea was discarded for two reasons. First, it would create a reconciliation issue. Secondly, it would create additional demands for storage space on the web server. A better solution would be to have the report in one place where both vendor and healthcare organization could access it. The reports subject to this multi-user access had to be placed in a directory structure that was not restricted using the file system security described above. The security enforcement would be migrated down from the directory level to the file level.

Embedding the authentication for the document inside the document solved this problem. The reports were all ASP pages, and thus enabled NCQA to imbed ASP code that could check the USERID of the person requesting the page. This is the same USERID used by the System Authentication process.

Extending the Database Access Control model from system access to file access was relatively simple.. The first step was to create an SQL Server lookup table (DocumentAuthority) with columns for USERID and FILENAME. Then, write a snippet of ASP code that would query an SQL Server table and return a row that contained the USERID and FILENAME involved. If no row was returned, the ASP code would redirect the user to the NCQA Secure Site Login page so that they could login correctly and view the page. Otherwise, they were allowed to view the report unhindered. See figure 4.

Figure 4



In this way, any number of users can view the same file, and the access to the file can be easily maintained and secured.

CONCLUSION

NCQA was faced with a problem created by growth. The data submission process had grown to a point where continuing the labor-intensive practices used in the past to manage the process was just unpractical. In addition, there was a clear mandate from the customers for faster turn-around time and better reporting capabilities. NCQA turned to a solution implementing a diverse assortment of technologies, including ASP, HTML, PDF, SQL Server, Windows 2000 Server (NT) and the SAS System. In this environment, SAS filled several roles. SAS was used to perform the statistical analysis, plus the statistical and descriptive reporting. SAS was also used as the DCS process manager. The DCS polling program was the implement of this process manager function. It controlled the actions of all the other technologies - calling them as they were needed, with appropriate parameters, and ensuring the proper synchronization of the entire DCS process. It was this ability of the SAS System to integrate the other discordant, but powerful,

technologies, into one cohesive system that enabled NCQA to meet the challenges of a secure online data validation and collection system for HEDIS survey data collection.

CONTACT INFORMATION

Christopher A. Roper
Manager, Information Systems
National Committee for Quality Assurance
2000 L St. NW
Suite 500
Washington, DC 20036
croper@ncqa.org
www.ncqa.org

Yong Li
Applications Developer
National Committee for Quality Assurance
2000 L St. NW
Suite 500
Washington, DC 20036
li@ncqa.org
www.ncqa.org

ACKNOWLEDGEMENTS & TRADEMARKS