Security Issues Through Authentication in Digital Content

C Krishna Kumar Poofa Gopalan

Abstract

While allowing the users to access the digital library from remote end, it is essential to provide security to the information stored in the host, as there are possibilities of changing the content of the digital library. Also it is essential to implement security measures to avoid the access by unauthorized users. The strategies followed in the field of information technology are Data security, Authentication and Cryptography. This area of IT deals with how the request from the user is identified as valid harmless. Also it provides techniques to avoid trapping the content on the way during its travel towards the destination.

0. Introduction

The following are some of the most important requirements of digital libraries.

- Adequate PCs with LAN connections.
- Local databases in machine readable form.
- Provisions to access CD ROM.
- Audio-Video library
- Vision network for observation purpose having vision cameras and TVs for strict vigilance.
- Access control system for automation entry of the user into library.
- Surveillance system at the exit gate which allows only the authorized documents to go out

Look at the last two requirements. These are the security part of digital libraries. We need systems to allow only authorized users to access the information. This spells the word AUTHENTICATION. Also we are in need of systems to supply only authorized documents to go out. This is achieved by cryptography. Even anybody can trap the information but they cant convey the real meaning as the information is totally changed into other means.

1. Authentication

Authentication usually involves a usemaine and a password, but can include any other method of demonstrating identity, such as a smart card, retina scan, voice recognition, or fingerprints. It is finding out if the person, once identified, is permitted to have the resource. This is usually determined checking if that person is a part of a particular group, if that person has paid admission, or has a particular level of security clearance. Finally, access control is a much more general way of talking about controlling access to a web resource. Access can be granted or denied based on a wide variety of criteria, such as the network address of the client, the time of day, the phase of the moon, or the browser which the visitor is using. Access control is analogous to locking the gate at closing time, or only letting people onto the ride who are more than 48 inches tall - it's controlling entrance by some arbitrary condition which may or may not have anything to do with the attributes of the particular visitor.

As digital libraries have content on the host that is sensitive, or intended for only a small group of people, the techniques in this paper will help make sure that the people that see those pages are the people that we wanted to see them.

The following are some of the authentication tools available in present web servers.

1.1. Basic authentication

As the name implies, basic authentication is the simplest method of authentication, and for a long time it was the most common authentication method used. We will discuss the basic authentication through an Apache web server.

1.1.1 How basic authentication works

When a particular resource has been protected using basic authentication, Apache sends a 401 Authentication Required header with the response to the request, in order to notify the client that user credentials must be supplied in order for the resource to be returned as requested. On receiving a 401 response header, the client's browser, if it supports basic authentication, will ask the user to supply a username and password, to be sent to the server. If you are using a graphical browser, such as Netscape or Internet Explorer, you will see a box, which pops up and gives you a place to type in your username and password, to be sent back to the server. If the username is in the approved list, and if the password supplied is correct, the resource will be allowed to access by the client. As the HTTP protocol is stateless, each request will be treated in the same way, even though they are from the same client. That is, every resource which is requested from the server will have to supply authentication credentials over in order to receive the content.

The browser takes care the details and hold it for that search session and there is no need to type in your username and password again. But you might have to type it again for next session for the same web site.

Along with the 401 response, certain other information will be passed back to the client. In particular, it sends a name which is associated with the protected area of the web site. This is called the realm, or just the authentication name. The client browser caches the username and password that you supplied, and stores it along with the authentication realm, so that if other resources are requested from the same realm, the same username and password can be returned to authenticate that request. This caching is usually just for the current browser session, but some browsers allow you to store them permanently, so that you need to never type in your password, for the next session.

The authentication name, or realm, will appear in the pop-up box, in order to identify the username and password.

1.1.2 Drawback of Basic Authentication

Basic authentication should not be considered as secure for any rigorous definition of security. Eventhough the password stored on the server in is encrypted format, it is passed from the client to the server in plain text across the network. Anyone listening with any variety of packet sniffer will be able to read the username and password clear as it pass.

The username and password are passed with every request, not just when the user first types them in. The packet sniffer will not be listening to the details at a particularly strategic time, it is listening to all requests coming across the wire.

In addition to that, the content itself is also going across the network in clear format. Hence if the web site contains sensitive information, the same packet sniffer may get access to that information, even the username and password were not used for a direct access.

Usage of basic authentication is not advisable for anything that needs real security. It is a detriment for most users, since very few people will take the trouble, or have the necessary software and/or equipment, to find out passwords.

1.2 Digest authentication

Another way of authentication is “digest authentication”, which is the improved technique of basic authentication. It is implemented by the module mod_auth_digest. There is an older module, mod_digest, which implemented an older version of the digest authentication specification, which will probably not work with newer browsers.

While using digest authentication, your password is never sent across the network in the clear, but is always transmitted as an MD5 digest of the user’s password. Hence the password cannot be determined by sniffing network traffic.

The full specification of digest authentication can be seen in the internet standards document RFC 2617, http://wwwl .ics.uci.edu/pub/ietlThttp/rfc26 17.txt and http://userpages.umbc.edu/ mabzugl/cs/md5/ md5.html

1.2.1 Drawback of digest authentication

Everthough digest authentication has great advantages, it is not supported by all major browsers and control. In particular, Opera 4.0 or later, Microsoft Internet Explorer 5.0 or later, Mozifla 1.0.1, Netscape 7 or later and Amaya support digest authentication.

Eventhough your password is not passing in the clear, all of your data is passing in the clear and hence this is a rather small measure of security. Hence your password is not really being sent, but a digest form of it is being sent. Hence somebody with their knowledge is workings of HTTP could use that information - just your digested password - and use that to gain access to the content.

The moral of this is that, if you have content that really needs to be kept secure, use SSL.

2. Database authentication modules

Basic authentication and digest authentication both suffer from the same major flaw. They use text files to store the authentication information. The problem is that, searching somethingfrom a text file is very slow. It rather like trying to find out something from a book without an index. We have to start working through one page at a time until we find what we are looking for. As we don’t remember the exact position of the same information in that file, we have to repeat the process every time.

Since HTTP is stateless, authentication has to be verified every time whenever the content is requested. Hence every time a document is accessed which is secured with basic or digest authentication, Apache has to open up those text password files and look through them on line at a time, until it identifies the user. In the worst case, if the username supplied is not in there, every line in the file need to be checked. On an average, half of the file will need to be read before the user is identified and this process is very slow.

This is not a big problem for small sets of users. But for larger number of users this becomes prohibitively slow. In many cases, valid username/password combinations will get rejected because the authentication

module had to spend longer time searching for the username in the file and Apache will get tired and return a failed authentication message.

In these cases, we need an alternative, and that alternative is to use some variety of database, which are optimized for looking for a particular piece of infonnation in a very large data set. It builds indexes in order to locate a particular record rapidly, and they have query languages for swiftly locating records that match a particular criteria.

There are numerous modules available for Apache to authenticate using a variety of different databases. The following are some of the modules which ship with Apache.

2.1 mod_auth_db and mod_auth_dbm

mod_auth_db and mod_auth_dbm are modules which let you to keep your usernames and passwords in DB or DBM files. There are few practical differences between DB files and DBM files. Some operating systems, such as various BSDs, and Linux, are exactly the same thing. You should pick the appropriate from these modules which makes the most sense on your particular platform of choice. If you do not have DB support on your platform, you may need to install it. You can download an implementation of DB from http://www. sleepycat. corn!.

2.2 Berkeley DB files

DB files, also known as Berkeley database ifies, are the simplest forms of databases, and are rather ideally suited for the sorting of data that needs to be stored for HTTP authentication. DB files store key/ value pairs of name of a variable, and the value of that variable. While other databases allow the storage of many fields in a given record, a DB file allows only this pairing of key and value. This is ideal for authentication, which requires only the pair of a username and password.

3. Conclusion

The various authentication modules provide a number of ways to restrict access to your host based on the user identification.

The access control mechanism allows you to restrict access based on criteria unrelated to the user identification. One of the major sources to connect the digital content host to the user is internet. We have seen how leading internet protocols and host servers incorporate the secured access and authentication. These are only the basic methods used for authentication. As the technology grows the methods of authentication may also be improved for better identification of the right user and proper security measures to the digital content or web content.

4. References

1. Recent Trends in Library & Information Science & Technology. Proceedings of papers presented in National Conference LIST 2002 at Bishop Heber College, Trichy, Tamilnadu, in 27—28 Jan 2002.
2. Future Libraries — S. Balakrishnan, P. K. Paliwal, 2001, Anmol Publications (P) Ltd., New Delhi, India.
3. Electronic Media and Library and Information Technology — Pandey. S. K. 2000. Anmol Publications (P) Ltd., New Delhi, India.
4. http://www. sleepycat. corn
5. http : 1. ics.uci. edulpub/ietf/http/rfc26 1 7.txt
6. http://userpages.umbc.edu/ mabzugl/cs/md5/md5 .htrn

About Authors

Mr. C. Krishna Kumar is student of 1st Year M.Tech, (Information Technology), Center for Information Technology and Engineering, Manonmaniam Sundaranar University, Tirunelvelli- 12.
E-mail : krishathi@yahoo.com

Ms. Poofa Gopalan is student of 2nd Year M.Tech, (Information Technology), Center for Information Technology and Engineering, Manonmaniam Sundaranar University, Tirunelvelli-12.
E-mail : abc_poofa@yahoo.co.in