

WEB CONTENT MANAGEMENT SYSTEM

February 2008

© The Government of the Hong Kong Special Administrative Region

The contents of this document remain the property of, and may not be reproduced in whole or in part without the express permission of the Government of the HKSAR.

Disclaimer: Whilst the Government endeavours to ensure the accuracy of the information in this paper, no express or implied warranty is given by the Government as to the accuracy of the information. The Government of HKSAR accepts no liability for any error or omission arising from or related to the use of the information.

TABLE OF CONTENTS

Summary	2
I. Introduction.....	3
What is a Web Content Management System?.....	3
Impact and Business Trends with WCMS	5
The Common Components of WCMS.....	6
II. Security Concerns and Precautionary Measures.....	8
III. Conclusion.....	11

SUMMARY

A Web Content Management System (WCMS) is a web application that facilitates a group of users, usually from different departments in an enterprise, to collaboratively maintain and organise the content of a website in an effective manner. Over the past few years, web content management systems have grown in importance as more and more organisations communicate and publish their information via the web. Like other web-based applications, WCMS's applications are exposed to the same set of common security threats found in any network and web-based operation or process. In this paper, we will outline the common security concerns of WCMS, and provide a number of precautionary considerations.

I. INTRODUCTION

WHAT IS A WEB CONTENT MANAGEMENT SYSTEM?

Since the dot-com boom of the late 1990s, corporate websites have become commonplace for almost any type of company, large or small, across the globe. Almost every enterprise these days needs a website to communicate with customers, partners, shareholders, and so on, providing up-to-date information on the enterprise, its products and services. Increasingly, commercial activities and order transactions are conducted on enterprise websites.

The Classic Approach to Web Content Updating

Building and setting up a website is not a one-time project. Different departments in the enterprise will have areas of content they need to add to and update. Plus, websites have to be maintained and updated on a regular basis due to the dynamic nature of modern business.

In the early days of website maintenance, the task of uploading and updating site content usually fell to the IT department. One method for uploading web content to the server was to use file transfer programs such as FTP (file transfer protocol). Another common approach was to create an upload function within a Web interface allowing different content owners to select appropriate files and upload them via HTTP. Both methods are common, and still used by web hosting companies and small & medium enterprises (SMEs).

Problems With the Classical Approach

Traditionally, technical staff would have to assist a content editor who needs to update a site by translating the content into a suitable web page format (i.e. HTML) and uploading it to the web server on their behalf. This iterative process often led to delays in publishing, and is obviously not an efficient process given the high mutual dependence required between the content provider and the technician.

Managing the website updating process is another problems with older approach. Sometimes a web page may consist of several content areas that require input and material from several different enterprise departments. When more than one person is able to update web pages simultaneously, the problem of logging and tracing “who has amended what” and “what the latest version of a page is” becomes serious.

Web Content Management Evolution

The Web Content Management Systems (WCMS) that have appeared more recently are designed to tackle these problems, and make it easier to collaboratively update a website. A WCMS is a web application that facilitates a group of collaborative users, usually from different departments across an enterprise, to maintain and organise web content in an effective and manageable way. Web content can include text, images, audio and video. A modern WCMS can also include workflow features so that the creating, storing, and updating of web pages, along with approval sub-procedures, can be streamlined. In addition, features such as versioning, check-in/check-out auditing, and so on are useful for managing and tracking the updating of web pages.

IMPACT AND BUSINESS TRENDS WITH WCMS

Commercial WCMS products have the following benefits¹:

1. Quicker response times: making new web content such as marketing materials available on the web is much quicker because content owners can update materials to a website directly, without the need to assign such tasks to technical personnel;
2. More efficient workflows: requests for changes and updates to a site are simplified under a WCMS framework. Users across different departments can add and apply changes to web content with a pre-defined and agreed-upon workflow process.
3. Improved security: under a WCMS framework, content is only published after approval by designated supervisors or managers. This reduces the chance of publishing material by mistake, which is usually due to human error. In addition, most WCMS systems provide audit trails of publishing activities all of which help maintain accountability;
4. Other benefits include improved version tracking, integration with translation servers, and consistency of page presentation through the use of common page layouts and controlled templates.

Web content management has grown in importance² over the past few years, and commercial as well as open source WCMS products are now available on the market.

¹ <http://www.edocmagazine.com/print.asp?ID=30578>

² <http://mediaproducts.gartner.com/reprints/fatwire/144978.html>

THE COMMON COMPONENTS OF WCMS

Many WCMS are programmed in languages such as Java and PHP, and run on a web server. In addition to the web server, WCMS may also contain additional components such as workflow engines, search engines, and email integration modules.

Web content and data is normally stored in data repositories or databases such as MySQL (open source) or Oracle (commercial). This could include text and graphic material to be published. Older versions of web pages from a particular site under management may also be stored in the database.

Generally, draft web pages are not uploaded directly to the production web server. Instead, users keep copies of draft pages offline until they are approved for publication. Then, once approved and signed-off, a file transfer program runs automatically, uploading and linking in the final pages on the production web server.

A WCMS is essentially a web application supported by a backend database, with other features such as search engine, and perhaps integration with a translation engine. The general security threats applicable to web applications, such as cross-site scripting, injection flaws and/or malicious file execution, can all be applied to a WCMS.

For the purposes of accountability, users normally need to be authenticated before they can access the WCMS. In some situations, users authenticate via an intermediate server called a reverse proxy server, instead of connecting directly to the WCMS server. In addition, content duties are segregated by dividing users into two groups—content editors and content administrators—where only content administrators have final publishing authority. The role of technical personnel would be in building web page templates and maintaining the consistency of web page layouts and a common look-and-feel.

Generally, data and content sent to a web server is considered public information. If it is necessary to store sensitive information on WCMS servers, appropriate data encryption and authentication measures should be put in place.

II. SECURITY CONCERNS AND PRECAUTIONARY MEASURES

As we have shown, a WCMS is an application built on top of existing web technology. Like other web applications, a WCMS is subject to the same security threats and operation process vulnerabilities as other web applications. In this section, we discuss the common security concerns and ways they can be mitigated.

Security Concerns

Given that a WCMS is a software application, it is prone to bugs just like any other program. Vulnerabilities have been found in WCMS. As one example, a vulnerability called “absolute path traversal vulnerability” was found in the open source product OpenCms in 2006. This flaw would allow remote authenticated users to download arbitrary files³.

Another security concern lies with protection of authentication credentials when accessing a WCMS. Many WCMS products are designed primarily to solve the content management problem of websites rather than building a secure product. Some WCMS products do not provide adequate protection for logins and passwords for example, and these passwords—including the administrator password—are sent as plain text over the network.

Similarly, as part of the publishing/uploading process, a WCMS might use file transfer protocols such as FTP to transfer files from the WCMS data storage server to the web

³ <http://nvd.nist.gov/nvd.cfm?cvename=CVE-2006-3934>

server. FTP is not a secure protocol in the sense that authentication credentials and passwords are sent as plain text over the network. In addition, because publishing is an automatic process from the WCMS to the production web server, FTP credentials might be hard-coded in certain configuration files. Usually a hard-coded login password like this will not be changed regularly. As a result, any leakage of this password could allow someone illegally access to web content on the production web server.

If the WCMS includes other modules, individual subsystems may have their own bugs and introduce their own vulnerabilities to the WCMS. For example, if the WCMS has an email module, it might be prone to the same common threats faced by email server such as email spoofing. On top of this, the backend database server of the WCMS may have its own vulnerabilities as well.

Precautionary Measures

There are a number of precautionary measures that should be done proactively to mitigate the security threats identified above:

1. Follow best practices by applying the latest security patches to all web server software. Any alerts or warnings about vulnerabilities on the WCMS product being used should be addressed immediately, especially if the WCMS can be accessed directly from the Internet. Any patch management process should also address additional WCMS modules, including email subsystems, backend database servers, JAVA runtime environments, and so on.
2. A strict password policy should be defined. This should include a minimum password length, initial assignments to personnel, restricted words and formats, and a limited password life cycle.
3. Logins and passwords sent over the Internet should be protected by SSL / TLS, so that attackers can't sniff them over the network. In general, access to

administration pages should be further controlled and these should not be open to Internet access.

4. When publishing any web content from the WCMS to the production web server, file transfer programs such as FTP should be replaced by a Secure Shell (or SSH) that protects transmission channels by encrypting data. Some SSH implementations also support a feature that controls which IP addresses are allowed to connect to the destination server.
5. To enforce data security, many WCMS implementations have built-in access control whereby groups of users are segregated into editor and administrator (approver) roles. These roles and their corresponding access rights should be clearly defined and reviewed periodically.
6. A good WCMS should keep an audit trail, logging all editing and approval activities. These audit trails should be retained for a period commensurate with their usefulness, and should be secured so they cannot be modified and can only be read by authorised persons.

III. CONCLUSION

While a good WCMS can facilitate businesses to better control their web content, making it more responsive in today's dynamic business environment, end-users should also be aware of the possible security impact on the enterprise if inappropriate material was published on the site. Advice to end-users include:

1. Be aware of what is being published. Only approved content should be involved in the publishing process.
2. Each user identity (user-ID) should represent only one person at a time. Shared or group user-IDs should not be permitted.
3. Passwords should be promptly changed if they are suspected of being, or have been, compromised or if they have been given to vendors for maintenance and support. Password management practices such as enforcing strong passwords, and regular changes of passwords should be followed.
4. Automatic protection features, such as a password protected screen saver, should be activated if there has been no activity for a predefined period to prevent any attempt at illegal system access.
5. When a member of a content editing and updating group ceases to provide services in that group or organisation, his or her WCMS user-IDs and access privileges should be terminated as soon as possible.
6. Software patches and updates should be applied to user machines regularly, including web browsers, Java runtime environments and so on, on a regular basis.