Kamlesh Kumar, Amarjeet Singh, Dr. Ashutosh Kumar Bhatt, Dr. Manoj Chandra Lohani / International Journal of Engineering Research and Applications (IJERA) ISSN: 2248-9622 www.ijera.com
Vol. 2, Issue 2,Mar-Apr 2012, pp.1391-1396

WEB CONTENT SECURITY SYSTEM OF DATA LEAKAGE

[1]Kamlesh Kumar, [2]Amarjeet Singh, [3]Dr. Ashutosh Kumar Bhatt, [4]Dr. Manoj Chandra Lohani

[1] Department of Computer Applications, Jai Arihant Academic Institute, Haldwani (Nainital)
[* 2] Department of Computer Applications, Institute of Environment and Management, Lucknow [*]
[3] Department of Computer Science, Birla Institute of Applied Science, Bhimtal,(Nainital) [*]
[4] Associate Professor, HOD, Department of Computer Science, Graphic Era Hill University, Bhimtal, (Nainital) [*]

ABSTRACT

Web application development is more complex and challenging than most of us think. But most Web-based systems are still poorly developed and in an ad hoc manner, resulting in poor quality and maintainability and contributing to failures. In this paper, we propose a web-based system for prevention of the confidential information leakage caused by the person who is authorized to access. This system realizes the centralized access control to the distributed confidential information and supports the confidential pages generated dynamically by web applications. We show the design and implementation of this system that is transparent to users.

Keywords: Web, Security, Information protection

1. INTRODUCTION

While the loss or leakage of data has become a major problem that needs to be solved in all kinds of organizations, the routes along which data can be lost have become complicated and numerous, making data loss countermeasure all the more difficult. Information leakage, detection and prevention (ILDP) is the new rising star in information security. For many years, the focus has been on the detection and prevention of intrusions. However, adequate measures must now be deployed to detect and prevent extrusions – compromises from within the organization. The risks posed by extrusions are clear and significant, yet most organizations are hampered today by the lack of solutions or expertise in the area of ILDP. Below are just some of the common examples of information leakage: Employees leaked key bid information to competitors unknowingly CEOs lost laptops or USB storage devices while in transit Employees, who are leaving the organization, copied competitive information to their personal email accounts More examples of data breaches can be found in the Chronology of Data Breaches of Privacy Rights Clearinghouse.

Many cases of information leakage go unreported due to fear of loss of confidence and regulatory penalties; hence, we are just looking at the tip of the iceberg. Information leakage can be caused by negligence or intentional sabotage. Emails are unintentionally sent to the wrong recipients. Besides negligence, it is a universal truth that the motivation to leak sensitive information will exist no matter what countermeasures your organization takes. And as storage media continually becomes more mobile and smaller in size, more sensitive information is likely to be stored on such media, having a greater likelihood of being lost or stolen. We have learned about information leakage incidents from history and we can be confident to see more of them in the future.

A web application is an application that is accessed via web browser over a network such as the Internet or an intranet. It is also a computer software application that is coded in a browser-supported language (such as HTML, JavaScript, Java, etc.) and reliant on a common web browser to render the application executable. Web applications are popular due to the ubiquity of web browsers, and the convenience of using a web browser as a client, sometimes called a thin client. The ability to update and maintain web applications without distributing and installing software on potentially thousands of client computers is a key reason for their popularity. Common web applications include webmail, online retail sales, online auctions, wikis and many other functions.A critical goal of successful information retrieval on the web is to identify which pages are of high quality and relevance to a user's query. Pages on the web contain links to other pages and by analyzing this web graph structure it is possible to determine a more global notion of page quality. Information finding is generally considered the dominant activity on the World Wide Web. In order to help users find information that is relevant to them, it is often necessary to understand the degree to which various entities on the web are similar or related to each other. As the Internet technology has become essential to office works, the office staff might deal with the confidential information such as customer information, personnel information, and designs of new products over the organization network. However, many incidents of confidential information leakage occur in organizations and most of these incidents are caused by the internal staff. Information leakage is one of the most serious security threats to the information security, but there are few effective methods to prevent it comparing with other conventional security threats such as eavesdropping of the network traffic.

In this paper, we propose the Data Leakage Prevention System (DLPS). This system provides protection of the confidential information stored in the web server against the information leakage such as bringing out the data by saving it as file, writing it to the media, and printing it out. Users can only read but cannot copy nor print the confidential information. The DLPS comprises four major components; Viewer, Encryption Proxy, Authentication Server, and Access Control Directory. Encryption Proxy, which is a proxy server interposed between client and web server, encrypts transmitted data of the confidential information

on demand. Adopting this encryption method, it is not necessary to change the existing web server that stores the confidential information, and the DLPS supports the confidential information generated dynamically by web applications such as CGI or Java Servlet. In addition, whenever Viewer accesses the confidential information, Authentication Server authenticates users and controls the access. The system administrator can manage the confidential information with the configuration of Access Control Directory and centralized access control of the distributed confidential information can be realized.

2. INFORMATION LEAK PREVENTION TECHNOLOGIES

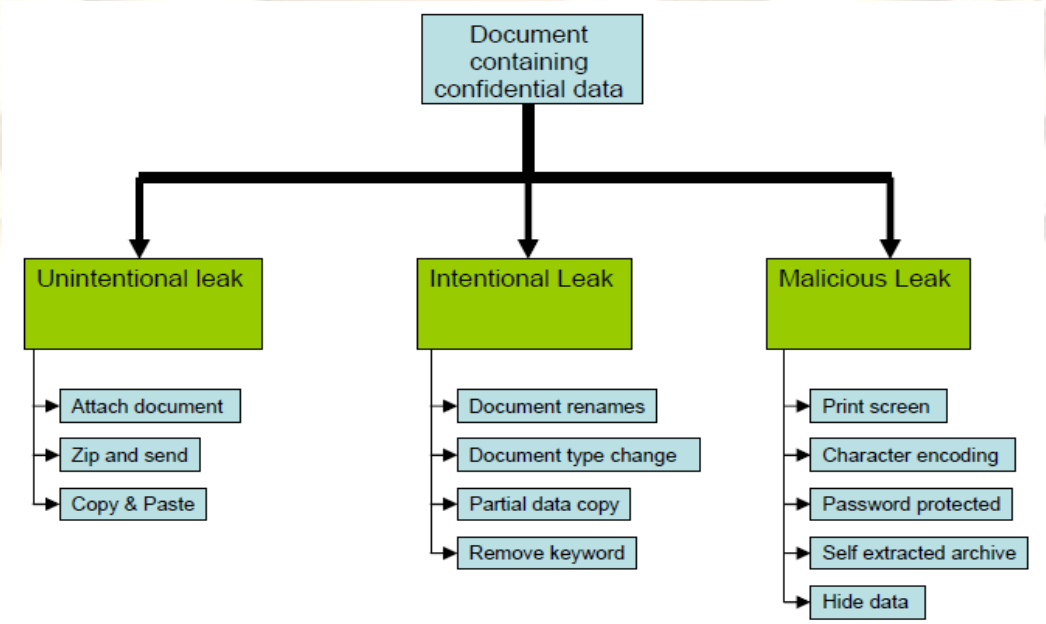
Information leaks occurs when users communicate from the "inside out", using a variety of different protocols such as e-mail, HTTP, FTP, Instant Messaging or even when printing confidential data. Product requirements to prevent leaks should include protocol agnostic capabilities (in other words, information leak prevention solution should prevent leaks for different outgoing communication protocols)

There are several technologies used by for content identification which Percept Technology Labs, Inc. tested. The three technologies tested by Percept Technology Labs for identifying sensitive information are:

- Keyword Content Filtering
- Regular Expression Content Filtering
- Precise ID Fingerprinting

When using keywords for identification, the leak prevention product scans the e-mail body and/or attachment for specific words or phrases – for example, scanning for the word "Confidential." When one of these words or phrases is found in the document or email body, a policy action is triggered and a resulting action is taken. A policy action may include, blocking or quarantining an e-mail and not allowing it to leave the company network. This technology is used by different vendors including PortAuthority Technologies, Symantec and CipherTrust.

Regular expressions are used to detect a specific order of digits and characters. For example, it can look for numbers in certain sequence. This technology is used by different vendors including PortAuthority Technologies, Symantec and CipherTrust products. PreciseID, invented by PortAuthority Technologies, identifies actual data rather than the presence of a keyword, or sequence of numbers inside a document. With PreciseID, the content of a protected document is scanned at rest, relevant data is extracted and a "fingerprint" representation of the data in the document is created. These fingerprints are then stored in a database and used to identify content in motion. PreciseID is designed to identify documents which are not exact match of the original document and may contain even small percentages of the protected data.



Classification of Information Leaks

Unintentional Leak – Leaking of data in which the person sending an e-mail did not intend to send the confidential data. Unintentional leaks can occur when a person is mistakenly attaching the wrong document to the email message or when Microsoft's Outlook auto completes the email address and the message is sent to the wrong recipient.

Intentional Leak – Leaking of information in which the person sending the e-mail is aware of his or her company policy but decides to try and send the confidential information anyway. Intentional leaks occur when the sender is aware of the policies and is bypassing security devices without trying to gain personal benefits. For example, when changing a document name or converting it to zipped archive.

Malicious Leak – Leaking of information in which the person sending the e-mail is deliberately trying to sneak information past the security product. Malicious leaks are very rare.

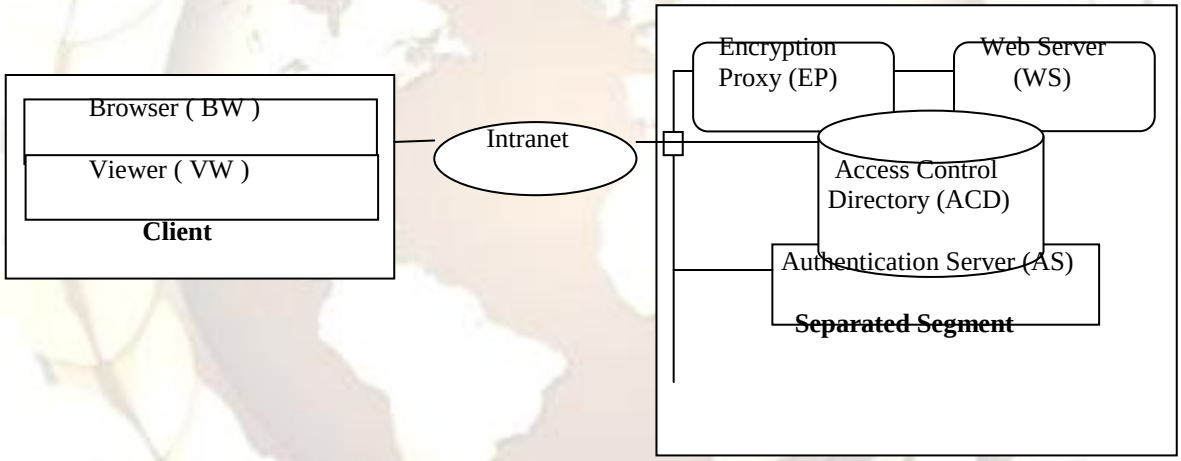
3. PRIOR WORKS

IBM research proposes a system for web content protection. This system provides digital rights management to off-the-shelf Web browsers and browser plug-ins. It verifies the browser code with the digital signature scheme and prevents users from performing actions that are not allowed such as print, save as, and so on. It distinguishes the protected contents with the specific protocol, and Internet Explorer invokes the Trusted Control Handler with these method names. Therefore, it is necessary to change the existing web pages to introduce this system. In addition, usage rights information is stored in the client and there is no mechanism to change the usage rights dynamically.

4. OUR APPROACH

We precise the architecture of the DLPS in Figure below. Web server, Encryption Proxy, Access Control Directory and Authentication Server are located in the separated network segment so that no one can access them directly. Furthermore, those components authenticate each other and the communications among them are secure. No one can access those components without being authorized.

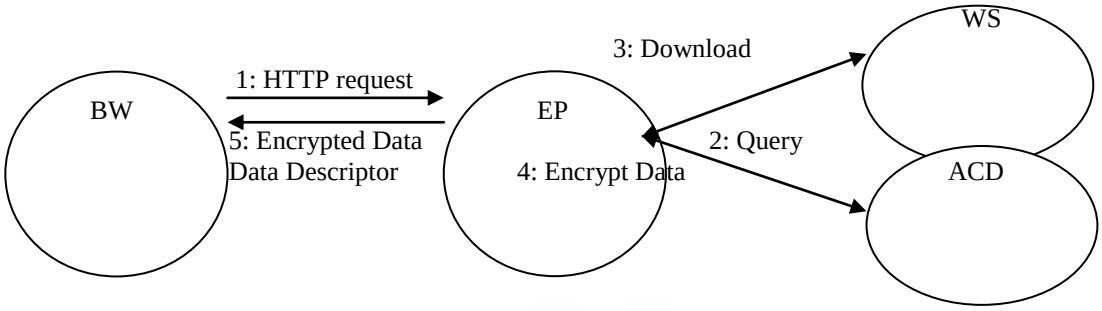
The web server stores confidential pages in plain form. Encryption Proxy is placed between the client and the web server to prevent the client from accessing to the web server directly via the network. It authenticates access users and encrypts the requested confidential page with the corresponding content key that is a secret key for encryption and stored in Access Control Directory. Viewer is installed in the client, which is an application to display confidential pages in this system. While Viewer is running, the user can neither take screen capture nor copy the displayed confidential information. Viewer has a function to save the encrypted confidential page.



System Architecture of Data Leakage Prevention System

Access Control Directory is the database in which administrator registers the following tables; user table and secret data table. The user table records user accounts and passwords. The secret data table records data descriptors, confidential page URLs, content keys to encipher and decipher the confidential pages, and access control lists. The user table is used in user authentication process. To check the access permission to each confidential page, the system uses the secret data table. When the user opens a confidential page with Viewer, it requests the content key of the confidential page to Authentication Server. Authentication Server queries access control list and checks permission of the corresponding confidential page to Access Control Directory with ID, password and data descriptor, which identifies the confidential page.

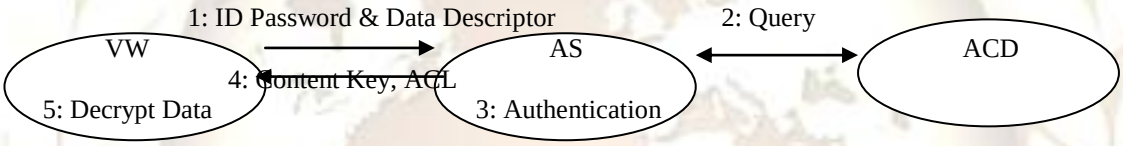
In order to control the access to the distributed confidential information, we design the dynamics of the DLPS that is composed of two phases; download phase and open data phase. The download phase occurs when the web browser sends a request for a secret page stored in the web server to Encryption Proxy (Figure below).



Download Phase of the System Dynamics

In the first step, the web browser sends HTTP request for a confidential page to Encryption Proxy. Encryption Proxy analyzes the HTTP request and queries Access Control Directory if the request is for the confidential page or not. If the request is for confidential page, Encryption Proxy downloads it from the web server and encrypts it with the content key stored in the Access Control Directory. After that, Encryption Proxy adds data descriptor to the encrypted data and sends it to the web browser. In this phase, user cannot see the confidential pages because they are encrypted and stored in the local client. Hence, Viewer needs to obtain the content key to decrypt and display the downloaded confidential page.

Open data phase is occurred when Viewer opens confidential pages encrypted and stored in the local client (Figure below). At first, Viewer sends ID, password, and data descriptor to Authentication Sever. After receiving those data, Authentication Server queries Access Control Directory in terms of ID, password, and data descriptor whether the user may access the confidential page or not. If user is allowed to see the data, Authentication Server sends content key to Viewer. Viewer decrypts confidential page data with the received key and displays it.

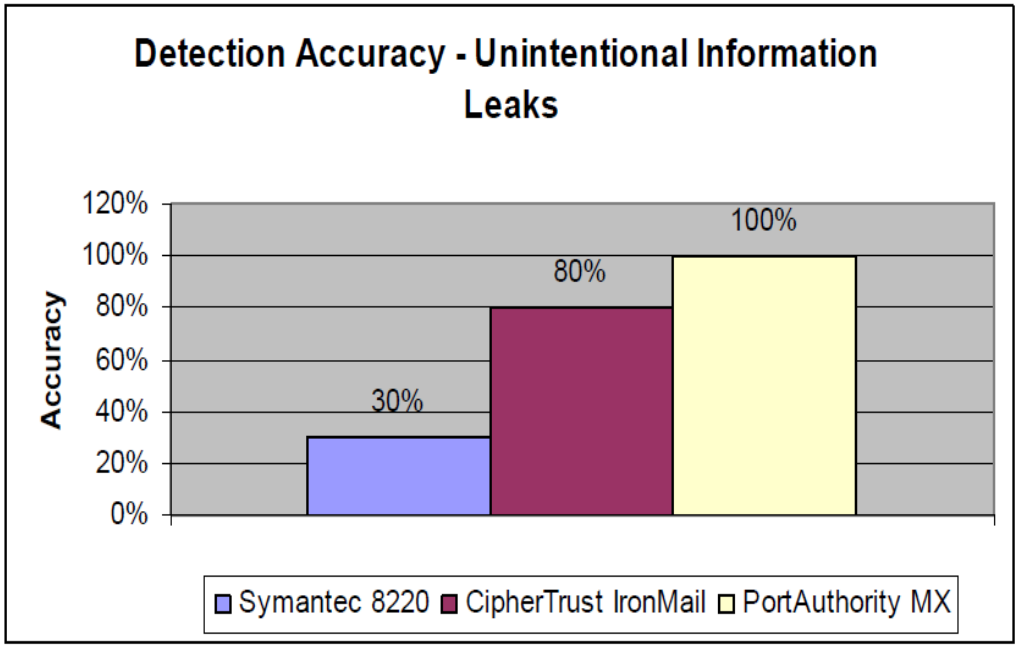


Open Data Phase of the System Dynamics

The most important process of the system dynamics is the two-way authentication between Viewer and Authentication Server. Actually, the result of user authentication is transmitted to Viewer precisely on the basis of the two-way authentication of those two components. Therefore, it is necessary for Viewer and Authentication Server to authenticate each other before the beginning of the session by authentication process such as the SSL mutual authentication. Applying this protocol, confidential page publisher can control the access to the distributed secret web pages, because users have to be authenticated and given access permission by Authentication Server whenever they open the confidential page with Viewer.

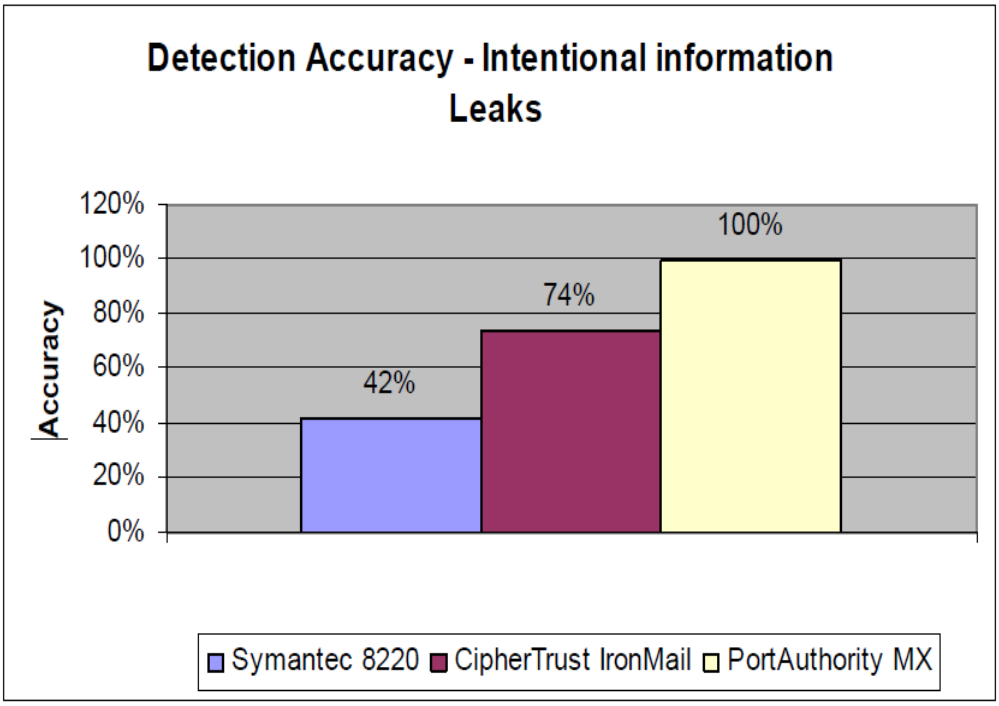
5. DESIGN AND IMPLEMENTATION

We built the prototype of the DLPS. The design concept of the prototype is transparency to users. Viewer is implemented as ActiveX control so that Viewer is embedded into Internet Explorer and the user is unaware of its running. If a component of frame page is a confidential page, Viewer control displays only a part of frame page as a confidential page on Internet Explorer. It restricts copy and print functions to prevent users from bring out the displayed confidential information.



DLPS Unintentional Detection Accuracy

Viewer control prohibits screen capture while it is running. To realize the prohibition of screen capture, it hooks and cancels the Win32 API methods by which capture tools take hardcopies. We can use the techniques to inject an arbitrary DLL into another process to hook the Win32 API methods. Using DLL injection technique, it is possible to manipulate the import section defined in each application and hook the Win32 API used in it. However, if the attacker could overwrite the import section of a capture tool after the injection of hooking DLL, it is possible to avoid the hooking routine and cancel the Win32 API hook. This is the imperative problem.



DLPS Intentional Detection Accuracy

The prototype of Encryption Proxy is the modified Squid proxy, to which we added the authentication function as plug-in of the Squid. On the arrival of the HTTP request issued by Internet Explorer, the Encryption Proxy analyzes the Proxy-Authentication attribute in the HTTP header field to authenticate the access user and returns a web page for Internet Explorer to activate Viewer control. Confidential pages' URLs and user accounts are previously registered in the Access Control Directory and the Encryption Proxy queries it with LDAP.

6. CONCLUSION

We proposed the DLPS that realizes the protection of confidential information. By the application of the super-distribution system architecture to the DLPS, it realizes the access control of the confidential information after its distribution. Furthermore, the on-demand encryption in the proxy realizes the support of confidential pages generated dynamically by web applications. We think that the DLPS is the essential architecture to deal with confidential pages and brings large effects to securities of the web system on the business.

REFERENCES

[1] C. Liu, and P. Albitz. "DNS and BIND (Fifth Edition)" O'Reilly Media Inc. May 2006.
[2] J. Schlyter and W. Griffin "Using DNS to Securely Publish Secure Shell (SSH) Key Fingerprints" RFC 4255. January 2006.
[3] R. Arends, R. Austein, M. Larson, D. Massey and S. Rose. "DNS Security Introduction and Requirements" RFC 4033. March 2005
[4] R. Arends. et. al. "Resource Records for the DNS Security Extensions" RFC 4034. March 2005.
[5] A. K. Ramani, R. C. Bunescu, R. J. Mooney, and E. M. Marcotte. Consolidating the set of know human protein-protein interactions in preparation for large-scale mapping of the human interactome. Genome Bi-ology, 6(5)-r40, 2005.
[6] M. Mourad, J.Munson, T. Nadeem, G. Pacifici, M. Pistoia, A. Youssef. WebGuard: A System for Web Content Protection. In Post Proc. of the Tenth International World Wide Web Conference, May 2001.
[7] A. Youssef. WebGuard: Making Copyright Protection Easy for Web Publishing. http://www-3.ibm.com/ibm/easy/eou_ext.nsf/ Publish/1829.
[8] Jeffrey Richter. Programming Applications for Windows Forth Edition. Microsoft Press.
[9] Super distribution: The Concept and the Architecture. http://www.virtualschool.edu/mon/ElectronicProperty/MoriSuperdist.html.