A Novel Approach for Protection of Confidential Web Contents

YASUHIRO KIRIHATA*, YOSHIKI SAMESHIMA*, TAKASHI ONOYAMA*, NORIHISA KOMODA**

*Research and Developing Department
Hitachi Solutions, Ltd.
4-12-7 Higashishinagawa, Shinagawa-ku, Tokyo 140-0002
JAPAN
{yasuhiro.kirihata.yk,yoshiki.sameshima.vf,takashi.onoyama.js}@hitachi-solutions.com

**Graduate School of Information Science and Technology
Osaka University
2-1 Yamadaoka, Suita, Osaka 565-0871
JAPAN
komoda@ist.osaka-u.ac.jp

Abstract: - Many incidents of confidential information leakage occur in enterprises and organizations. Most of these incidents are caused by internal malicious staff. Confidential information leakage is one of the security issues but there is no effective method to prevent so far. In this paper, we propose a web content protection system for prevention of information leakage, which protects the confidential information stored on the web server. Applying this system to the conventional web system, system administrators can manage the access control of the confidential information and prevent them from being leaked out from the office. We describe the system architecture and evaluate the security of the system implementation.

Key-Words: - Data loss prevention, Web security, Content access control, BAN logic

1 Introduction

As the Internet technologies are spread into commercial industries, accumulating and exploiting information through a LAN become essential requirements under highly competitive business environment. Sharing office documents on the network brings large efficiency to the office works. Users might deal with the confidential information such as customer information, personnel information, and designs of new products in the LAN. In order to protect the network from the outside attacker, a lot of techniques and products have already been developed. For example, setting up a firewall between Intranet and Internet, the network administrator restricts and audits the network traffic and keeps attackers out of the Intranet. To protect from the intrusion attack against the web server, the network administrator can detect the attack with the Intrusion Detection System (IDS) [1][2] or construct the web site on the Secure OS such as Security Enhanced Linux [3] and OpenSolaris with Solaris Trusted Extensions [4]. In addition, innovative studies are performed to protect the system from not only these direct attacks, but also indirect attacks, for example Denial of Service Attacks (DoS) and Distributed Denial of Service Attacks (DDoS) [5][6].

However, it becomes a serious problem that malicious people disclose confidential information in companies and organizations. Most of these incidents are caused by the internal staff of the organizations that can access the confidential information regularly. These crimes are increasing gradually as the propagation of the Internet and it is difficult to prevent them by the conventional system. Enforcing appropriate security policies for system operations to users is only one of the efficient ways to prevent these crimes so far, but it is difficult to enforce security policies completely.

In this paper, we propose the Web Content Protection (WCP) system in which user can only read but cannot leak out confidential information stored on the web server. The WCP system comprises four major components; Viewer, Encryption Proxy, Authentication Server, and Access Control Directory. Encryption Proxy, which is a proxy server set up between client and web server, encrypts transferred data of the confidential information on demand to support the dynamically generated web contents. The
system architecture realizes the simple installation to the conventional web system without any modification of the target sites.

2 Security Issues on the Conventional System

Before discussing the WCP system, we consider the security issues of the conventional web system that deal with confidential information. Figure 1 shows the possible attacks to the conventional web system on the LAN. As shown in this figure, there are several threats to overcome for the protection of the confidential information on the web server.

![Figure 1. Threats on the conventional web system](image)

(1) Spoofing: Attackers might spoof the web server as authorized user to steal the secret pages. To spoof the web server, attackers usually crack password. For instance, they sniff the network (online password cracking) or analyze a local password file (offline password cracking) and steal the ID and password of the authorized users to access the web server. Generally, one-time password or robust password management are the efficient defensive measures against these attacks.

(2) Eavesdropping: Sniffers are the tools with which network administrators observe the network communication for troubleshooting. There are several implementations widely available through the Internet such as TCPDump, EthDump, and Packetman. However, using sniffers, attackers can intercept the transferred data on the LAN. They can analyze the transferred data and steal the confidential information or ID and password of the authorized user to spoof the web server.

(3) Sneaking: Usually, authorized users can access the secret page on the web server and save it in the client machine as necessary. If they bring out the saved secret page, the confidential information is leaked. The sneaking causes most of the incidents of information leakage. From the viewpoint of the information leakage, it is essential to defend the web system from sneaking.

(4) Screen capturing: Capture tool is a tool to copy whole or a part of screen data as image data. It is available on the Internet as freeware or shareware. After downloading secret pages and viewing the secret information with browser application, the authorized users might take the capture image of the screen with capture tools and sneaking it as an image data.

(5) Intrusion: The goal of most system attacks is intrusion, that is, attackers get the 'root' privilege of the target machine so that they could operate the victim machine freely. To succeed in the intrusion, attackers might carry out buffer over flow attacks or crack the 'root' user's password. If the web server was invaded, confidential information on the web page might be stolen easily.

As for the eavesdropping, spoofing and intrusion, there are a lot of methods to defend the system. For example, installing the IDS into the web server, we can detect the intrusion attacks. To construct the web site on the Secure OS, remote user cannot get the 'root' privilege and it becomes difficult for remote attacker to intrude the web server. Against eavesdropping, encrypting transferred data is an effective method. It is possible to avoid spoofing attacks by applying one time password system or changing password periodically.

However, it is difficult to avoid sneaking and screen capturing the secret information by authorized users. In the conventional web system, authorized user can copy or print out the displayed confidential information by using ordinary functions of web browser. The goal of the WCP system is to defend the web system from sneaking and screen capturing.

3 System Architecture

3.1 System Requirements

Before illustrating the architecture of the WCP system, we describe the requirements of the system.

(1) Compensate for vulnerabilities of the conventional web system

As we discussed in the previous section, we need to defend the web system from several vulnerabilities. To prohibit the information leakage, the WCP system has to cover them completely.

(2) Manage the access control for the distributed confidential information

After distributing the secret data, the system administrator might want to manage the access control
for the distributed secret data in accordance with the change of access rights for some users. For instance, if a user who had the access right to a secret page and saved the secret page data to the local client machine, the administrator of the WCP system needs to change the access rights of the saved secret data when the user moves to other department or its role is changed.

(3) Support to secret pages generated dynamically
In many companies, the confidential information, such as customer information and personnel information, is stored in databases. When user queries the database system to search the confidential information, web application such as CGI and Java Servlet dynamically generates the result page. Accordingly, the WCP system needs to support the secret pages generated dynamically.

### 3.2 Architecture of the WCP system
Under the above requirements, we propose the WCP system. Figure 2 illustrates the system architecture of the WCP system. The system comprises web server, Encryption Proxy, Access Control Directory, Authentication Server and client. Viewer is installed in the client, which is an application to display secret web pages in this system. In addition, we assume that Viewer is a tamper-resistant application. Thereby, it is difficult to analyze and modify the binary code of Viewer. Web server, Encryption Proxy, Access Control Directory and Authentication Server are separated physically so that anyone cannot access them directly and they identify mutually. Furthermore, we assume that the communication in the separated network segment is secure and no one can access each component without being authorized. The web server stores secret pages with plain text. Encryption Proxy is placed between client and web server to prevent the client from accessing to the web server directly via the network. We explain each component in detail.

(1) Viewer
Viewer downloads an encrypted secret page and negotiates Authentication Server to get a content key, which is a secret key for encryption. When Viewer negotiates Authentication Server, it requires user to enter ID and password. After the negotiation, Viewer decrypts the downloaded secret page with the content key and displays it. While Viewer running, the user can neither take screen capture nor copy the displayed confidential information. Viewer can save the encrypted secret page.

![Figure 2. System architecture of the WCP system](image)

(2) Encryption Proxy
This component analyzes the HTTP request sent from the client and queries Access Control Directory whether it is for a secret page or not. If the HTTP request is for the secret page, it downloads the secret page from the web server and encrypts it with the corresponding content key and sends it to the client.

(3) Access Control Directory
This is the database in which system administrator registers the following tables; user table and secret data table. The user table records user account and password. The secret data table records data descriptor, secret page URL, content keys to encipher and decipher the secret pages, and access control list. The user table is used by authentication of access users. To check the access permission to each secret page, the system uses the secret data table.

(4) Authentication Server
When the user opens a secret page with Viewer, it requests the content key for the secret page to Authentication Server. Authentication Server queries access control list and checks permission of the corresponding secret page to Access Control Directory with ID, password and data descriptor, which identifies the secret page.

### 3.3 Internal Process of the System
We explain the internal process in the WCP system. On the WCP system, we can divide system dynamics into two phases; download phase and open data phase.

(1) Download phase
Figure 3 shows the data flow among the components of the WCP system on the download phase. This phase occurs when Viewer sends a request for a secret page stored in the web server to Encryption Proxy.
On the first step, Viewer sends HTTP request for a secret page to Encryption Proxy. Encryption Proxy analyzes the HTTP request and queries Access Control Directory if the request is for the secret page or not. If the request is for secret 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 Viewer. On this phase, user cannot see the secret web pages because the secret web pages are only encrypted and stored in the local client. Hence, Viewer needs to obtain the content key to decrypt and display the downloaded secret page.

(2) Open data phase

This phase is occurred when user opens secret web pages encrypted and stored in the local client. Figure 4 illustrates the data flow among Viewer, Authentication Server and Access Control Directory on the open data phase. 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 secret page or not. If user is allowed to see the data, Authentication Server sends content key to Viewer. Viewer decrypts secret page data with the received key.

4 Implementation

The prototype of the WCP system consists of Viewer, Encryption Proxy, and Access Control Directory. We have implemented the prototype of Viewer on Windows and Encryption Proxy on FreeBSD.
(1) Seamless GUI of Viewer to web browser
   When the user accesses to secret pages with web browser, Viewer is automatically started and displays the secret page into the web browser. The user does not aware of the existence of Viewer as if Viewer is the plug-in module of the web browser. He can see the secret page like ordinary pages. Furthermore, he can move to another hyper-link page on Viewer as well as web browser.

(2) Distribution and version management of Viewers
   Operating the WCP system, the administrator has to distribute Viewers to each client machine and manage their version. Therefore, distribution and version control mechanism are essential. We implemented Viewer as ActiveX control to realize that.

(3) Prohibition of screen capture
   While Viewer running, user may not take screen capture with capture tools or the Print Screen key. It is relatively easy to prohibit the Print Screen key by cancelation of the Print Screen keyboard event. However, prohibition of the screen capture is not simple. To realize this, we implemented prohibition mechanism of the GDI function calls for screen capturing.

(4) Reutilization of existing components
   The Internet Explorer is constructed from several components. WebBrowser control is one of those components and a HTML parsing and rendering engine of the Internet Explorer. Reusing this component in Viewer, it is easy to maintain the HTML parsing and rendering function of Viewer.

5 Security Evaluation with BAN Logic
   As we described in section 3.3, the system dynamics can be divided into two phases; download phase and open data phase. In this section, we discuss the validity of the system dynamics on the viewpoint of the BAN logic [10][11].

   In the system dynamics of the WCP system, which is the essential part for authentication? Unless an attacker cracked Encryption Proxy or modified the Access Control Directory, downloaded secret pages are automatically encrypted and no one can see the secret information until the finish of the download phase. It is clear that there is no weakness of the protocol in the download phase because of its simplicity. Therefore, user cannot see the secret pages before being authenticated by Authentication Server in the open data phase. To discuss the vulnerability of protocol in the WCP system, we only need to consider the open data phase.

   Figure 6 shows the idealized protocol of the authentication protocol in the open data phase.

   ![Figure 6. Idealized open data phase in the BAN logic](image)

   In this phase, the assumptions are that (1) the Viewer and the Authentication Server share the secret key $K_0$, (2) the Viewer believes the Authentication Server generates the valid session key $K$, and (3) the Viewer and the Authentication Server believe their own nonce and key to be fresh respectively.

   \[
   \begin{align*}
   &V_w \not\in V_w \not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\not\n
Above all, we can conclude the following four statements.

\[ \text{AS} \in \text{Vw} \Leftrightarrow \text{Vw}^K \Rightarrow \text{AS} \quad (10) \]

These four statements are the goal of the verification of this protocol's validity in the BAN Logic. Hence, we can conclude that open data phase of the protocol is secure.

6 Related Works

Digital Rights Management (DRM) technology allows digital content owners to distribute valuable digital content with its copyright not infringed by malicious people and set the accessible duration and conditions. The DRM technology will be applied to the industries of music and e-books. However, it does not support to the web contents dynamically generated by web applications.

IBM research proposes a system for web content protection [8]. It verifies the browser code with the digital signature and prevents users from performing several actions such as printing, saving, and so on. It provides transparent DRM functions to an application. However, it distinguishes the protected contents with the specific protocol such as rmfile (for local content) and rmhttp (for remote content). The Internet Explorer invokes the Trusted Control Handler with these method names. This implies that it is necessary to change the existing web pages for this system. In addition, usage right information is stored in the client and there is no mechanism to change the usage right dynamically. Hence, it is impossible to change the access rights of contents dynamically for roles and users.

7 Conclusion

The confidential information leakage becomes one of the most serious security problems in enterprises and organizations. In this paper, we proposed the WCP system that realizes the protection of confidential information on the web server. Our approach prohibits copy of the displayed text by the mouse operation and taking hardcopy of the displayed image when the confidential data is on the browser window. In addition, adopting the on-demand encryption method in the proxy, the WCP system can support the confidential information generated dynamically by web applications such as CGI and Java Servlet, and it is the distinctive feature of our solution. According to the security evaluation with BAN logic, we concluded that the internal protocol of WCP system is secure and it realizes the secure web distribution framework.

References: