



Professional Information Security Association

Phone : (852) 8104-6800
Fax : (852) 2900-8338
Email : info@pisa.org.hk
URL : www.pisa.org.hk

Professional Information Security Association (PISA)

Hong Kong E-Commerce Security 2003

December 2003

Version 1.0

DISCLAIMER

The study was conducted in a legal, ethical and non-intrusive manner and complied with the code of ethics stated in this report. The material and discussion in this study are solely for security awareness promotion and educational purposes. This material is NOT intended to be adopted in the course of attacking any computing system, nor does it encourage such act.

Furthermore, the points made here are deliberately kept concise for the purpose of presentation. If you require technical details please refer to other technical references.

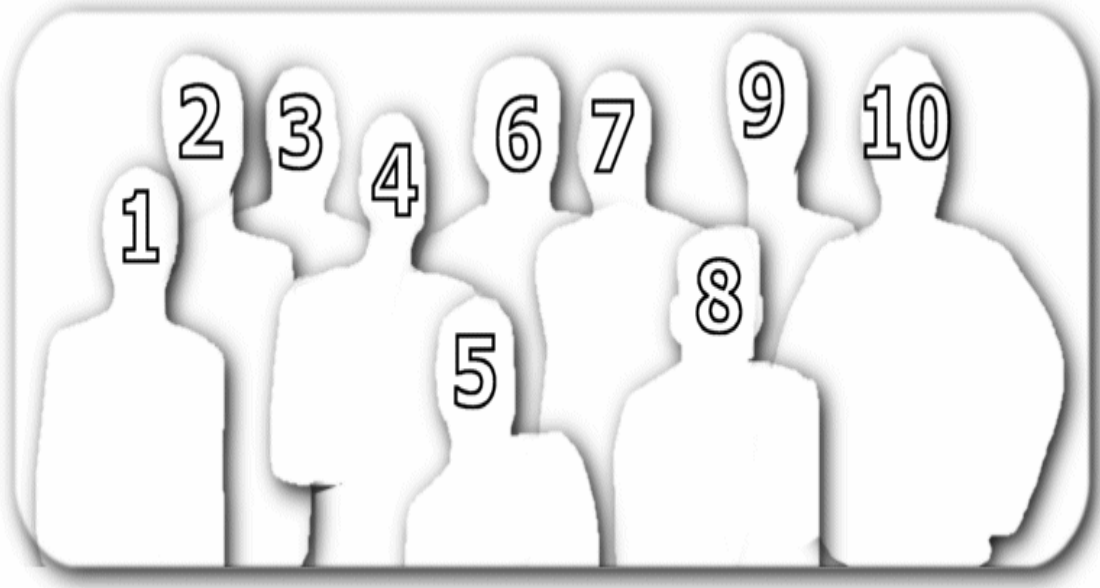
PISA would warn that unauthorised access to computer system, damage of data and computer system are offences.

PISA takes no liability to any act of the user or damage caused in making use of this report.

COPYRIGHT

The copyright of this material belongs to the Professional Information Security Association (PISA). A third party could use this material for non-commercial purpose, given that no change in the meaning or interpretation of the content was made and reference is made to PISA. All rights are reserved by PISA.

PROJECT TEAM MEMBERS



(1) Benjamin CHAN (2) John CHEUNG (3) Alan HO (4) Howard LAU (5) Raymond LEE
(6) SC LEUNG (7) Patrick LIU (8) Raymond LO (9) Billy NGUN (10) Leo SIN

CODE OF ETHICS

The organiser and investigator of the study agreed on the following points to the study to take care of the security and privacy issues.

- The objective of the E-Commerce Security Survey was to study the technical and operational aspects of e-commerce web sites security in Hong Kong, and to arouse the public awareness in securing the e-commerce web sites.
- We employed non-intrusive methodology in our study to discover the vulnerabilities. We did not penetrate into any web site to further explore their vulnerability; we did not intercept any third party network traffic nor did we jam any network traffic.
- We do not publicize or infer the name of individual vulnerable web site or the owner. We publicize only anonymous information and consolidated figures.
- We limit to the scope we state above only.

EXECUTIVE SUMMARY

Although the Internet bubble has burst for a few years and many Internet start-ups have gone bankrupt or disappeared, the e-commerce transactions and Internet activities do not show any sign of slump. All these can be seen from the increase in Hong Kong's broadband access and Hong Kong's high e-business readiness ranking given by the Economist Intelligence Unit.

Given that information security is crucial to the development of e-commerce, PISA would like to have a dissection on the information security status of the Hong Kong e-commerce technical infrastructure in the year 2003. Unfortunately, there are not many published articles or papers that can be found in Hong Kong depicting the situation. All these motivated PISA to conduct a more comprehensive e-commerce security study (the "study") in Hong Kong.

The project was started in February 2003 and ended in October 2003 with 10 security professionals who are PISA members participating in the study. 25 e-commerce web sites from a variety of industries in Hong Kong were studied. 17 of them are from financial industry and 8 of them are from non-financial industry. The aim of the study is not to draw any conclusion about the web security level of Hong Kong as a whole or any particular sectors, but to highlight areas found to be inadequate and propose practices that can be adopted to improve the situation.

The study was conducted in an ethical, legal and non-intrusive approach. A total of 45 questions covering 3 aspects, namely, infrastructure, application and operation, were used. In the infrastructure aspect, the study examined the DNS zone transfer setting, web server software version & service patch level, X.509 certificate management and server SSL configuration of the web sites. In the application aspect, the study looked into the items related to web page, session and password management. In the operation aspect, the study took a look of the control procedure related to the web site operation and transaction. Common and easily available tools, such as Internet Explorer, Netscape browser and basic network commands, are used to facilitate the study.

DNS zone transfer, which is originally used to replicate DNS information to multiple name servers for fault-tolerance and network performance consideration, is often used by hackers to map a company's Internet network topology. The study found that some companies failed to restrict the zone transfer. The study also found that not every web site had their web server version and patch level up-to-date that would give hackers chances to exploit well-known vulnerabilities. SSL is now a baseline technology used by web sites to provide confidentiality and authentication between the web browser and web server. But the study found that there were 12% of web sites did not employ SSL in the web sites for login or making transactions. For those employed SSL, some of them failed to properly show the SSL padlock at the lower right hand corner of the browser on the login page. The SSL padlock is an indicator for end users to easily recognise the use of SSL in a web site and to authenticate the web site through verifying the digital certificates before proceeding the login or online transaction.

Although most of our sample web sites supported high grade encryption and secure SSL version, the study found that some still support low grade encryption and less secure SSL version. As SSL allows “no encryption but MAC only” mode, one of the questions is to find out if any web site supports this insecure mode. The study found that the answer of this question was YES.

X.509 certificate is a vital component of SSL. The study included a few questions on X.509 certificate management. The study found that all the certificates provided by the sample web sites were within their validity period and their certificate chains were complete. However, some of the web sites used X.509 version 1 certificates which were out-dated and unable to provide necessary security protection such as Certificate Revocation List Distribution Point (CRLDP) and basic constraints. On the other hand, some web servers failed to provide CRLDP even though their X.509 certificates are in version 3.

Regarding the application aspect, the study took a quick look of the content of some web pages (say, the login page) in order to see if they contained sensitive information and comments. The results showed that some web sites need to do more to filter out sensitive information and comments from the web pages before production. Besides, they have to pay more attention to the randomness of the session ID, automatic time-out of a login session, disabling back page and browser history feature, and the use of cookie in a secured manner.

Another area in the application aspect is password management. The study reviewed the sample web sites based on good password practices, such as adopting strong passwords, disallowing password reuse, periodic changing of password and forcing to change password on first time access. The results found that there are rooms for improvement in these areas.

Finally in the operation aspect, the study found that some companies required face-to-face authentication during account opening. However, it is less common for the second authentication, such as entering password or digital signature before approving a transaction. During the study, the project team noticed some exceptions that were not in our scope. These included leaking of database structure during SQL exception handling and leaking of user ID and password because of the use of HTTP GET method.

The study report has provided a number of recommended security practices that can be deployed by the companies. We would highlight that new security challenges will appear from time to time. It is essential that the government, corporate management, security professionals, and the users have to work together to meet the challenges.

TABLE OF CONTENTS

1	Introduction and Objectives.....	1
2	Methodology.....	2
3	Findings.....	4
3.1	<i>Infrastructure Aspect.....</i>	4
3.1.1	DNS Zone Transfer Setting.....	4
3.1.2	Web Server Software Version and Service Patch Level	5
3.1.3	X.509 Certificate Management.....	6
3.1.4	Server SSL Configuration	9
3.1.5	Interesting Finding.....	11
3.2	<i>Application Aspect</i>	12
3.2.1	Web Page Management	12
3.2.2	Session Management	12
3.2.3	Password Management	15
3.2.4	Other Exceptions	17
3.3	<i>Operation Aspect.....</i>	18
3.3.1	Control Procedure.....	18
4	Recommended Practices.....	19
4.1	<i>Infrastructure Aspect.....</i>	19
4.1.1	DNS Zone Transfer Setting.....	19
4.1.2	Web Server Software Version and Service Patch Level	20
4.1.3	X.509 Certificate Management.....	20
4.1.4	Server SSL Configuration	22
4.2	<i>Application Aspect</i>	23
4.2.1	Web Page Management	23
4.2.2	Session Management	23
4.2.3	Password Management	25
4.2.4	Others.....	25
4.3	<i>Operation Aspect.....</i>	27
4.3.1	Control Procedure.....	27
5	Future Challenges.....	28
6	Reference.....	29
7	Appendix.....	31
7.1	<i>Hong Kong E-Commerce Security 2003 Checklist.....</i>	31
7.2	<i>Survey Statistics</i>	34
7.3	<i>Top 10 Security Issues.....</i>	36

1 Introduction and Objectives

Although the Internet bubble has burst for a few years and many Internet start-ups have gone bankrupt or disappeared, the e-commerce transactions and Internet activities do not show any sign of slump. According to the Office of the Telecommunications Authority (OFTA), as of September 2003, the Internet traffic volume through broadband networks is 100,289 terabits which is four times more than the traffic volume one year ago. October 2003 statistics of OFTA also indicated that the number of registered broadband customer account, which was around 1.2 million, surpassed the number of registered dial-up account. It is expected that consumers who have broadband buy more online than narrowband users. In April 2003, another survey on the e-business readiness rankings done by the Economist Intelligence Unit gave Hong Kong's overall ranking at the tenth which is the highest ranking ahead of many other Asian countries.

As the security professional, we would probably ask if the web sites are doing enough in protecting their customers over e-commerce when embracing this trend. With the awareness promotion and endeavour of the security professional, firewalls, operating system security and intrusion detection system become the commonplace and baseline safeguards against the Internet threats. Although these elements are still critical components of any security infrastructure, they are clearly insufficient to stop a new generation of attacks that are increasing in frequency every day now. Firewalls, operating system security, and the latest patches can all be bypassed with a simple attack against a web application.

Given that web application security is crucial to the development of e-commerce, we would like to know the current situation in Hong Kong. Unfortunately, there are not many published articles or papers that can be found in Hong Kong depicting the situation. A survey done in 2001 by AC Nielsen on the number of secure servers per 100,000 persons aged 16 years and over with Internet access shown that the US led the world with 49 secure servers while Hong Kong ranked tenth with only 15 secure servers. However, this survey doesn't tell us the situation in other Internet security areas, such as password management and X.509 certification management. All these motivated PISA to conduct an e-commerce security study (the "study") in Hong Kong. The aim of the study is not to draw any conclusion about the web security level of Hong Kong as a whole or any particular sectors, but to highlight areas found to be inadequate and propose practices that can be adopted to improve the situation.

2 Methodology

Addressing security, privacy and legal concerns

Our first hurdle was to design the survey such that it could address to all security, privacy and legal concerns. We spent quite some time in figuring out the approach. Complement with the approach is the code of ethics which all investigators needed to comply with.

Investigators of the Survey

Ten PISA members were involved in the project as investigators. They are information security professionals in the field so that the quality and the compliance to security practices and ethics were guaranteed. All investigators were briefed with the objective and code of ethics of the research and had to commit to abide by them.

Selection of Sample Web Sites

Twenty-five e-commerce web sites of a variety of industries in Hong Kong were studied. Generally speaking, 17 of them were from the financial industry (such as banks, securities firms, insurance companies and other financial-related companies) and 8 of them were from non-financial one. The web sites were selected based on several criteria. Firstly, most of them were providing online transaction services. Secondly, our investigators should have a legal user account to access the user functions of the web site, allowing the investigator to collect information without infringing the security or privacy of other party. Finally they needed to have certain degree of local popularity.

Benchmarking

We benchmarked our findings with the industries' common and best practices in web site information security. These findings might provide indication of potential security threats to the web sites. However, since we were not involved in the design and implementation the web sites, we could only point out general issues as reflected from our finding and might not be able to tell if any flaws have been offset by other mitigation measures which we could not have observed in our limited scope of tests.

We understand that no web site is perfect and security holes could never be totally. With reference to some common and best practices in web site information security, we attempted to make recommendations to minimise the level of security threats and the chances of attacks.

Publication of Findings

We would not disclose the identity of the web sites that we studied. Even though we provide some statistics, they were presented in a way that is basically unable for readers to infer the identity of the web sites.

Tools Used

Common user tools like Internet Explorer and Netscape Browser were used for viewing the certificate and SSL settings of the web sites. A personal proxy tool was used to view the HTTP header, cookies, session ID, web page content and data items at the client side.

Test Performed

A non-intrusive approach was adopted. Only general data that flew in and out of a common Internet browser was collected and analysed. In addition, general network information was captured using basic network troubleshooting commands like “nslookup”. We did NOT perform intrusive attack to applications like buffer overflow attack, SQL injection, brute-force password attack, privilege escalation, session hijack and spoofing attack.

We focused on areas where mis-configuration, mis-use or improper management might become vulnerabilities in the leakage of sensitive or confidential information, e.g. application and client information. In the infrastructure aspect, we examined the DNS zone transfer setting, web server software version & service patch level, X.509 certificate management and server SSL configuration of the web sites. In the application aspect, we looked into the items related to web page, session and password management. In the operation aspect, we took a look of the control procedure related to the web site operation and transaction.

Because of limited resources and time, we used a “short-cut” method for some checking items. For example, when we checked the randomness of the session ID, we just checked the length and changes in data content of 5 consecutive session IDs in a short period of time rather than doing more detailed analysis of the session ID patterns for more session IDs over a longer period of time. Hence, there is a limitation in our judgment of the randomness.

Phases of Study

The project was kicked off in February 2003 and ended in October, 2003.

a) *Initiation Phase*

- i) *Scope, Objectives and Code of Ethics* – These were first things that were discussed in the project team. The team fully understood the importance and necessity to proceed the project in compliance with the Code of Ethics.
- ii) *Team Formation* – 10 PISA members who were interested in the project formed the project team.
- iii) *Site and Question Selection* – E-commerce web sites in Hong Kong with online transaction services were eligible for study. Also, a non-intrusive set of checklist questions were brainstormed and set up for pilot checking.

b) *Pilot Phase*: the pilot phase objectives were to verify the validity of the checking tools and procedure; to make sure that each investigator understand the process and align them to the same standard. 10 sample web sites were selected. The investigators were divided into 5 sub-teams, each of which was allocated 2 web sites to test with. Investigators in the same team cross-checked each other’s results and marked down irregularities of the results and the procedure. After the first round of test, a review was conducted a review to refine the checklist and procedure.

c) *Live Test Phase*: with the new procedure, each team were assigned new web sites. They run the tests on the 2 old web sites and newly assigned web sites.

d) *Analysis Phase*: the project team gathered all the results from the groups and then analysed the findings, clarified unclear points and discussed interesting findings.

3 Findings

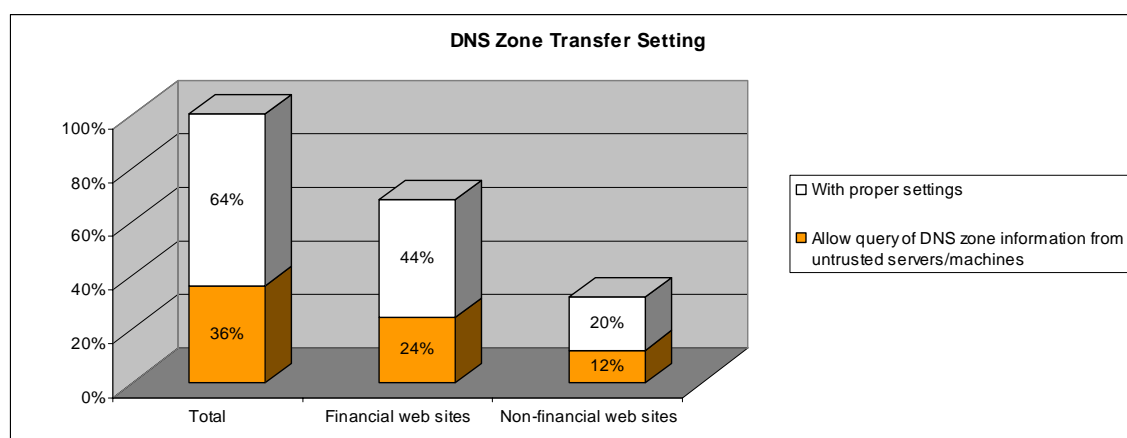
3.1 Infrastructure Aspect

3.1.1 DNS Zone Transfer Setting

DNS (Domain Name Service) is an important service that resolves domain names of network equipment/devices into their corresponding IP addresses, or vice versa. DNS resource records of a company can provide hints of the servers and network infrastructure of a company. The domain/host names, IP addresses and types of resource record serves as a roadmap to a hacker in locating which and what machines/devices to attack. Thus, it would be dangerous if more than necessary DNS information is revealed to the public.

Replication of DNS zone information by “zone transfer” to multiple DNS servers is common for fault-tolerance and performance considerations. DNS zone transfer, however, should be controlled with proper authentication between authorised servers, and if possible, in encrypted format. In a recent study of the UK’s DNS Infrastructure by Network Penetration, it mentioned some UK second level domains allowed zone transfers and raised the concerns of exposing zone information to untrusted hosts [NP].

In our study, we found that 36% (24% from financial industry and 12% from non-financial industry) of our checked web sites allowed untrusted parties to obtain a detailed list of DNS information by using basic network troubleshooting commands. Some corporations have their own DNS name servers while others subscribe to DNS name hosting services by third party providers. It is recommended to review and restrict the DNS zone transfer setting where appropriate for both internal and third party provided name servers.



3.1.2 Web Server Software Version and Service Patch Level

There are a variety of choices of web & application servers and databases that run on a variety of operating systems. Security loopholes due to software program bugs or mis-consideration could happen in whatever class of software tools and platforms. Regular upgrade and application of security patches is vital in keeping the system robust from unwanted attacks. Software of older versions are more likely to be attacked since the information of security loopholes and exploits have been known to the public for a longer time and also the level of support from the software vendors is getting less or even unsupported.

As of October 2003, the followings were the latest versions of some popular web servers:

Web Servers	Reference
Apache 2.0.x	http://httpd.apache.org
Microsoft IIS 6.0	http://www.microsoft.com/WindowsServer2003/iis/default.msp
Sun Java System Web Server 6.1 (Sun ONE Web Server / Netscape Enterprise)	http://www.sun.com/software/product_categories/web_servers.html
IBM HTTP Server 2.0.x	http://www-306.ibm.com/software/websevers/httpsevers/
Zeus 4.2 release 3	http://www.zeus.com/products/zws/

By examining the HTTP headers or “banners” sent back by the server, we collected the Server Header information from our sampled web sites. Based on the Server Header information, the following is a brief summary of the web server software version and service patch level of our checked web sites:

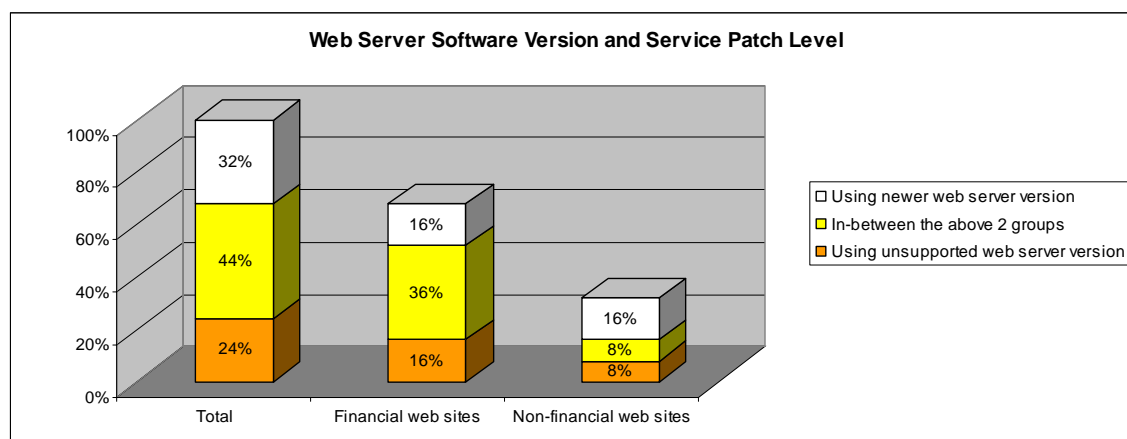
Web Servers	Total %	Financial Industry %	Non-Financial Industry %	Still Supported by Vendors?	Using newer versions?
Apache 1.3.x (below 1.3.26)	16%	12%	4%		
Apache 1.3.x (on or above 1.3.26)	24%	8%	16%		Yes
Microsoft IIS 4.0	20%	12%	8%	No	
Microsoft IIS 5.0	8%	8%	0%		Yes
Netscape Enterprise 3.6 SP2	4%	4%	0%	No	
Netscape Enterprise 4.0 or 4.1	20%	16%	4%		
IBM HTTP Server 1.3.x (below 1.3.26)	8%	8%	0%		

(Remarks: Please note that the Server Header shown on the HTTP headers might not be the true version numbers of the web servers since such information could be altered by some tools or settings. Hence, the actual situation could be a bit different from our findings.)

The following were some news related to the level of support on the web servers:

- Netscape Enterprise version 3.6 has been declared not supported by Netscape Product Management team in April 2000. [Netscape]
- Microsoft has announced retirement of Windows NT Server 4.0 (i.e. including IIS 4.0) in September 2001. [Microsoft]
- IBM mentioned that Program Temporary Fixes (PTFs) V5R2 for iSeries servers (that was released in September 2002) is the final release that supports the original HTTP web server. IBM intends to make most of the future enhancements to HTTP Server (powered by Apache 2.0.43) and asks customer to consider migration of the HTTP server (original) to the new version. [IBM]

In the above, we found that there were 24% (16% from financial industry and 8% from non-financial industry) of the web servers of our checked web sites are not supported by vendors any more. It is recommended to upgrade these web servers in the near future. On the other hand, we found that there were 32% (16% from financial industry and 16% from non-financial industry) are using newer versions of the web servers.

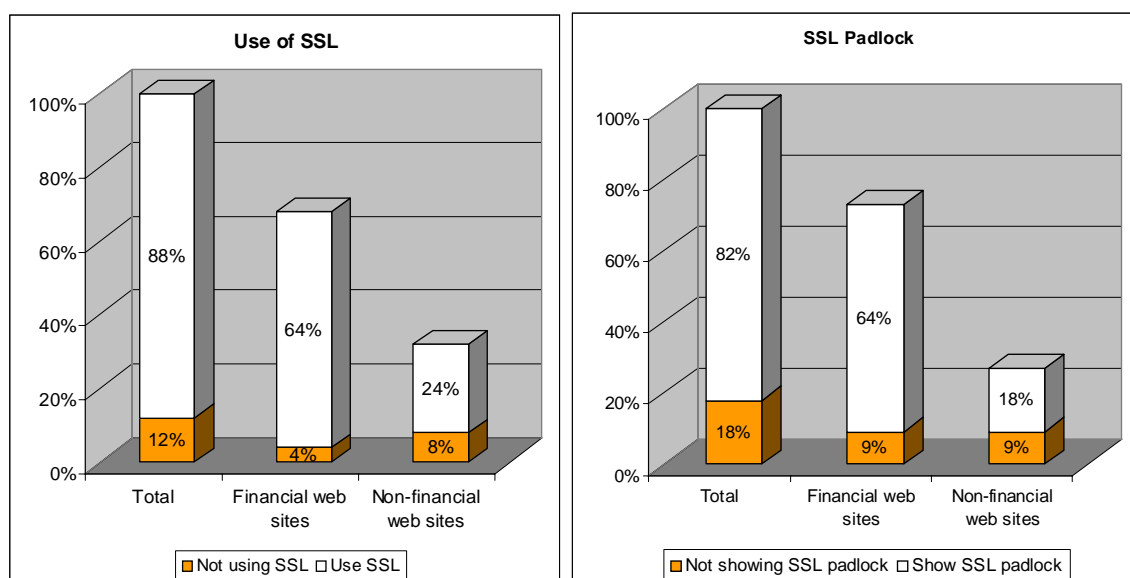


3.1.3 X.509 Certificate Management

Data transfer by HTTP is in plain text format which is not secure. Hackers can easily intercept and modify the HTTP content between web server and end users. Alternatively, hackers can also set up fake web sites to capture credit card number and password information if the end users did not verify the identity of the web sites or their security certificates before login. Several fake financial web sites reported recently in the mass media were good examples. Thus security mechanism is needed for authentication of web site as well as transport of sensitive data and information. The use of SSL (secure socket layer) transport protocol, together with X.509 certificate is the answer to both requirements.

SSL was developed by Netscape in 1994, employing X.509 Certificate technology as platform to exchange secret keys for building an encryption channel between web server and end users. SSL is now the industry standard for secure transport of Internet communication. It is supported by all the major web browsers. End users do not need to install additional plug-in or program in order to support SSL web site.

88% (64% from financial industry and 24% from non-financial industry) of our samples applied SSL to protect sensitive data (user name, password, transaction details, etc) and provide authentication to their web sites. However, some poorly configured web sites failed to clearly indicate that the session is encrypted or activate SSL before the login session/web page. For web sites using SSL, 18% (9% from financial industry and 9% from non-financial industry) failed to show the SSL padlock in the lower end corner of the browser. SSL padlock is a mean to enable users to quickly recognise that the communication is encrypted and verify if the web site's URL address matches with the one indicated in the X.509 certificate issued by CA. It was found the URL of one web site failed to match with the identity on its SSL server certificate.



Another important element in X.509 technology is the Certificate Authority (CA). Web browsers will recognise well known root CAs. Unrecognised CAs will be regarded as invalid and result in errors. All of our samples used valid CA and valid certificate applied from valid CAs. All the certificates were within their validity period. The following table shows the distribution of the CA used by the web sites:

CA	Percentage
Verisign	60%
RSA Secure Server	8%
Thawte	16%
Hong Kong Post	4%
Without using CA	12%

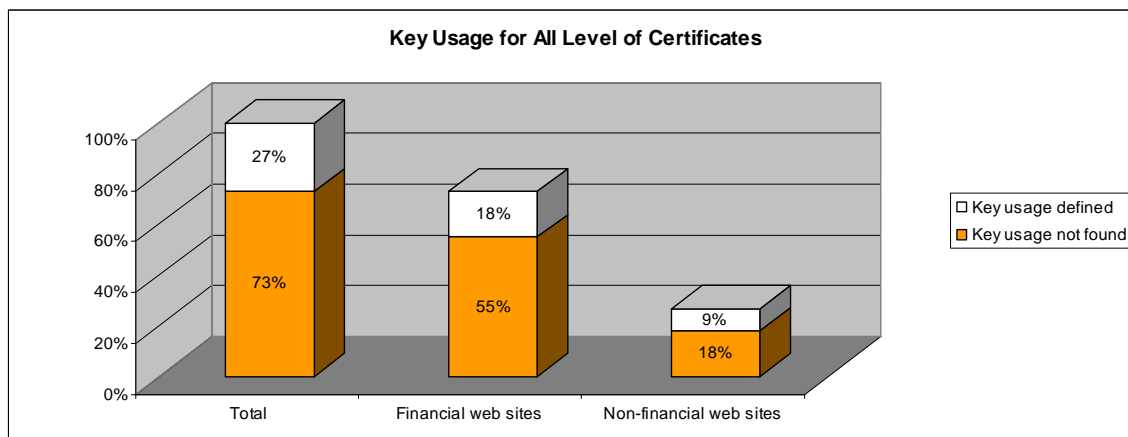
We found that all web server certificates were of X.509 certificate version 3, except one which was of version 1.

We noted that some of the CAs did not issue SSL server certificates directly using their root certificates. These SSL server certificates were issued using intermediate CA (or sub-CA) certificates that were issued by the root CA certificate. This forms a 3-level certificate chain. Excluded those without SSL, our result indicates that 87% (64% from financial industry and 23% from non-financial industry) of the web servers provided full certificate chains to the browsers and they were all valid. It also indicates that root certificate in version 1 is quite common.

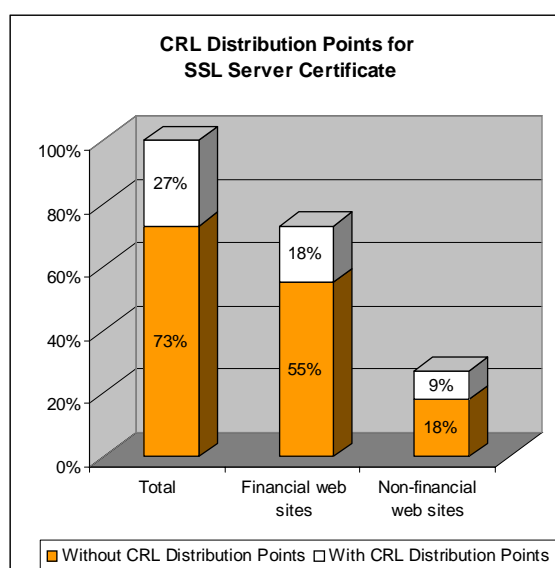
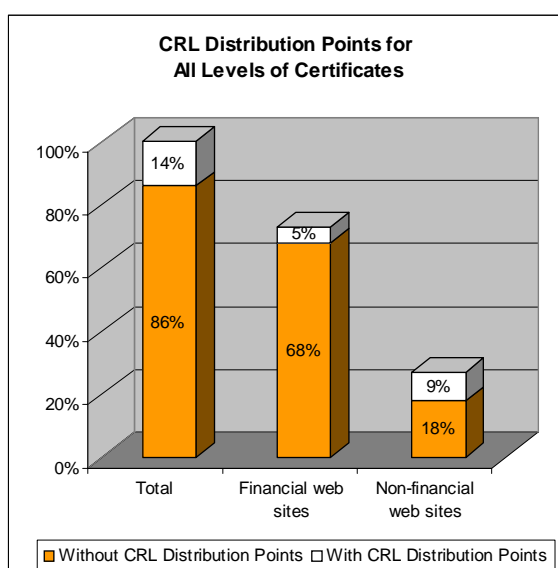
“Basic Constraints” is a critical extension field in X.509 Version 3 certificate specification. A SSL server certificate cannot be used to issue other SSL server certificates if its “basic constraints” is set to be an “end-entity”. In July 2002, there was an exploit to IE 5.x and 6.x using the basis constraints. IE failed to spot fake SSL server certificates that were signed by another SSL server certificates instead of a CA certificate. For details of this exploit, please refer to Microsoft Security Bulletin MS02-050 and PISA’s workshop “*To Trust or Not – SSL Security Vulnerabilities*” [PISA-SSL]. Most of the SSL server certificates’ basic constraints were properly set. However, there is a

web server which was still using version 1 SSL server certificate, intrinsically possess no basic constraints field. This SSL server certificate could be used to exploit the above mentioned IE vulnerability if a browser has not been properly patched.

Other key usage fields are optional settings such as “Server Cert Enhanced Key Usage” and “Key Usage” were implemented respectively by 82% (64% for financial and 18% from non-financial industry) and 27% (18% from financial industry and 9% from non-financial industry) of our samples which implemented SSL. They give extra usage constraints on the SSL.



CRL (Certificate Revocation List) is a mechanism for distribution of compromised certificates. This feature enables end user to check the validity of a specified SSL certificate. Only 14% (5% from financial industry and 9% from non-financial industry) of our sampled SSL web sites were come with this setting for all levels of certificates. If we counted only the SSL server certificates, 27% (18% from financial industry and 9% from non-financial industry) of the web sites were with CRL.

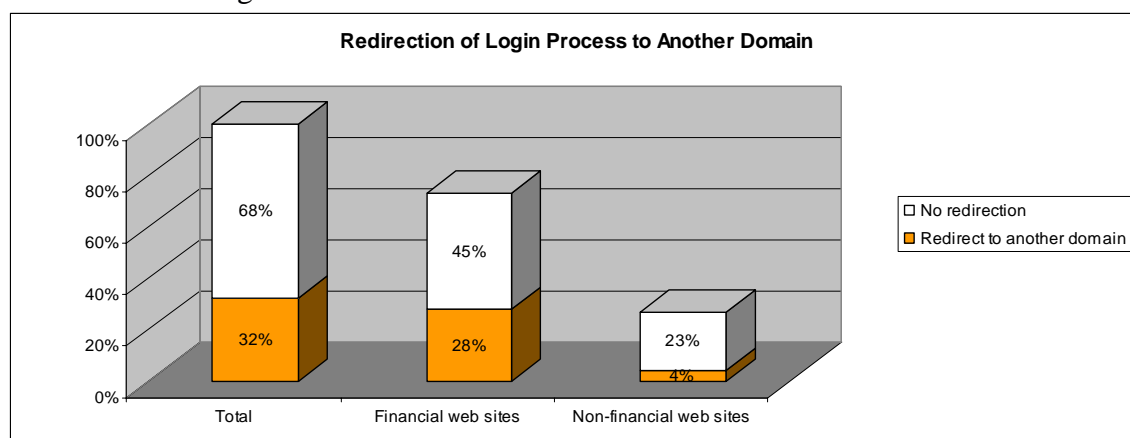


Key length is the critical measurement of the secure level of SSL certificate. Result shows that most of the key lengths of the certificates are 1024 bits. 19% (5% from

financial industry and 14% from non-financial industry) of the CA certificates were with key length of 1000 bits which is less secure in today's standard. We found that there was one server certificate with key length 512 bits which is well below standard.

A successfully authenticated web site requires a match of the server certificate and domain of the web server. If a web site is poorly configured or designed to have submission of information to another domain (i.e. a different SSL connection was initiated), an alert dialogue box will pop up. 32% (28% financial industry and 4% from non-financial industry) of the SSL web sites redirected the login process to another domain. Although the different domains were probably owned by the same company, the alert dialogue box would confuse the user and set a bad user experience of the security of the web server.

It is insecure to redirect login information to a non-SSL URL. None of our sample was found to submit login information to non-SSL URL.



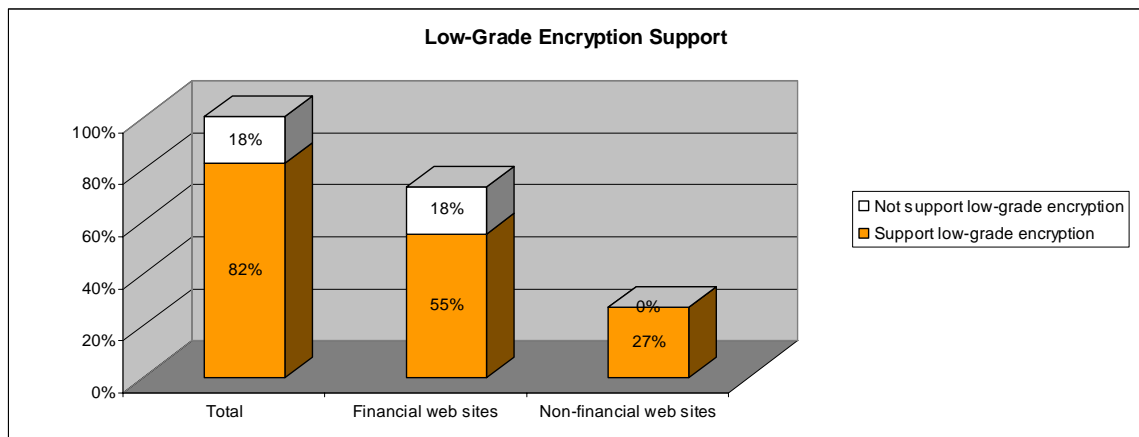
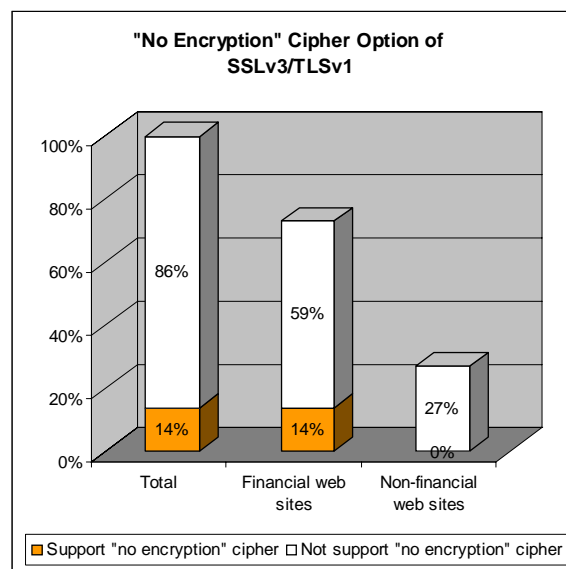
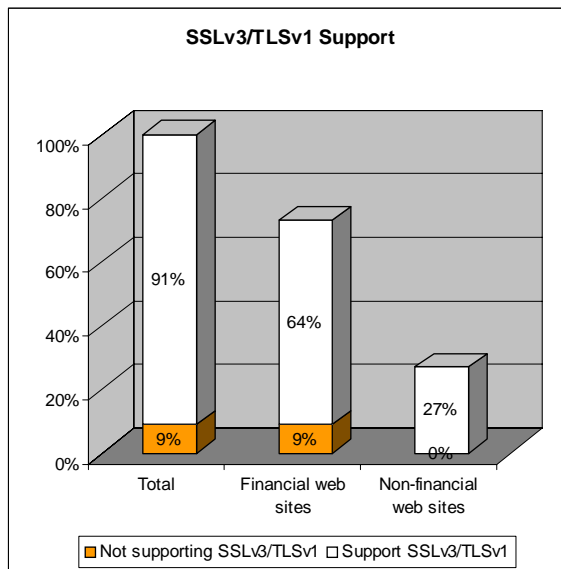
3.1.4 Server SSL Configuration

Although SSL connection initiation is controlled by the end user, the web server can limit some insecure features by appropriate server settings. When end users start SSL connection negotiation with web server, the web server will provide the ciphersuite and encryption key length for the SSL connection. Exception the 3 web servers (1 from financial industry and 2 from non-financial industry) without SSL, all other 22 web servers with SSL enabled were using high-grade ciphersuites for SSL connection.

The following statistics in this section are calculated based on those with SSL features. 95% of them used RC4-128 and only one of them used Triple DES. One of our questions is to check whether SSLv2, which is insecure, was still supported by the web sites. We found that 72% (45% from financial industry and 27% from non-financial industry) of them still supported SSLv2. It may be due to downward compatibility considerations. However it was worth to note that 9% of SSL enabled sites (all from financial industry) did not support SSLv3 or TLSv1.0. They provided only less secure SSLv2.

SSLv3 has a feature which allows no encryption for the SSL connection which is not secure. In our studies, 14% (all from financial industry) of the SSL web sites allow no

encryption on the SSL. In year 2003 when AES was not yet popular in browsers, the strongest ciphersuite to be supported by most browsers in SSLv3 or TLS1.0 connections is 3DES with a 168-bit key and a SHA-1 MAC. 91% (the remaining 9% are from financial industry) of the SSL web sites supported the strongest ciphersuite. 82% (55% from financial industry and 27% from non-financial industry) of the SSL web sites supported some low-grade encryption which only has 40-56 bits key length. Only 18% (all from financial industry) of the web sites specifically do not support any low-grade encryption key length at all. One interesting phenomenon arose from the result of this question was that 28% of the SSL web sites did not support RC4 40 bits or RC4 56 bits encryption but these web sites supported other low-grade encryption, such as DES 56-bits.

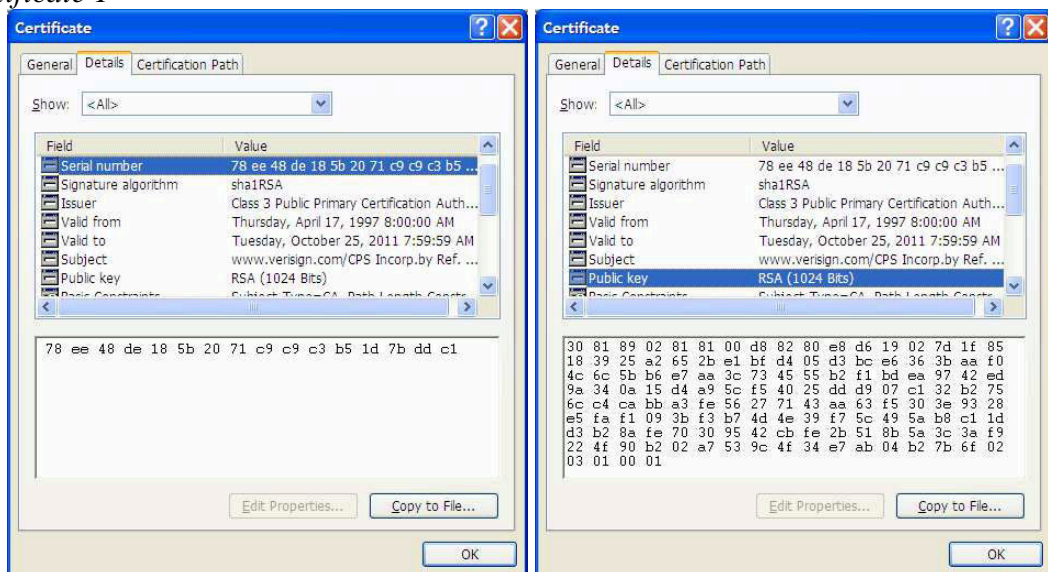


3.1.5 Interesting Finding

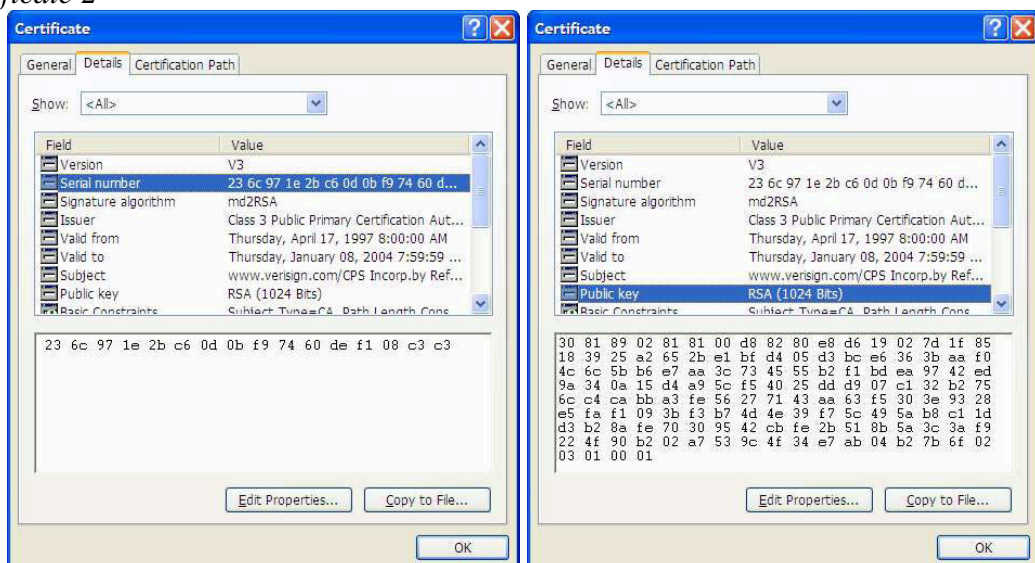
3.1.5.1 Different intermediate CA certificates with different serial numbers but with the same public key

During our checking, we found 2 different intermediate CA certificates with different serial numbers but with the same public key. The two intermediate CA certificates are both with subject “www.verisign.com/CPS Incorp.by Ref. LIABILITY LTD.(c)97 VeriSign”, however, they are of different signature algorithms, different enhanced key usages and different valid dates. They are both issued by “VeriSign Class 3 Public Primary CA” (see below).

- *Certificate 1*



- *Certificate 2*



According to the specification of RFC 3280 [RFC3280],

- A certificate contains the distinguished name of the certificate issuer (the signer), an issuer-specific serial number, the issuer's signature algorithm identifier, a validity period, and extensions.
- A Certificate Revocation List (CRL) lists the serial numbers of revoked certificates whose validity has prematurely revoked. The date on which the revocation occurred is also specified.

Since certificate revocation is based on the “serial number”, a CA can revoke a certificate by registering the “serial number” of the problem certificate to the CRL. We just wonder what would happen if the public key was compromised.

3.2 Application Aspect

3.2.1 Web Page Management

A web page (HTML page) may contain displayable and non-displayable items. Displayable items could be a table column heading or some display data items. For non-displayable items, they could be some hidden data fields, program comments, JavaScript programs, etc. These can be viewed simply using the “view source” function of a browser.

In a poorly designed web site, web pages may contain sensitive hidden data fields or program coding that could be viewed via a browser. Furthermore, a hacker may even alter the data content of sensitive hidden data fields during data passing to the web server that might affect the processing results of the web application.

In our study, we found a financial web site with many hidden data fields and some of them were sensitive since they seemed to be related to login information and encryption parameters.

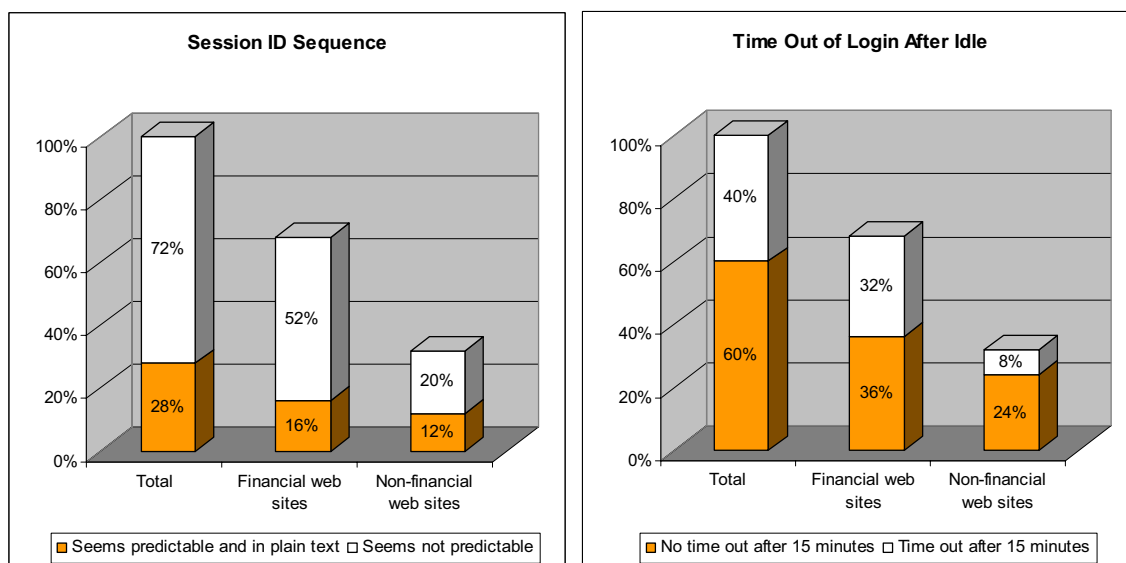
We also found a financial web site with many internal program comments that explain the processing logic. Even worse, some of item validation logic of this web site is written in JavaScripts that are run in a client browser. Thus, the program source of the JavaScripts can be easily downloaded to the client PC and viewed. It could be a security issue since some parts of the login handling logic is exposed.

3.2.2 Session Management

There were a number of issues we studied under Session Management, including (1) system time leakage, (2) session ID sequence, (3) session time-out, (4) “no-cache” policy, (5) “back” page allowance, (6) “history” feature after logout, and (7) Cookie.

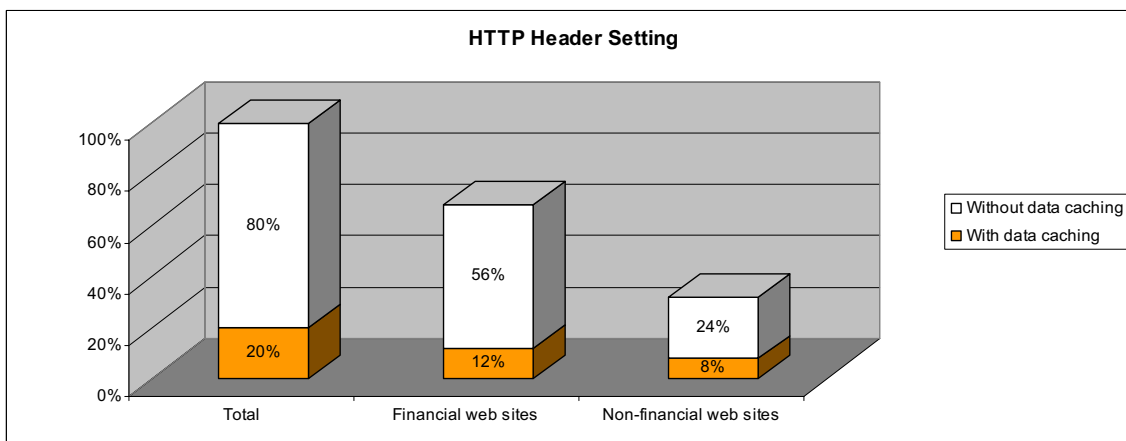
For nearly all studied web sites, their server leak real system time information. It seemed that it is unavoidable to show the system time of the server as a proof of the time of transaction.

In our study, we found that about 28% (16% from financial industry and 12% from non-financial industry) of our checked web sites showed some levels of predictable sequence ID in plain text. Predictable sequence ID was not just about simple numeric increment (e.g. 23456, 23457, 23458...). In fact, we found some Session IDs with increasing numbers or alphabets. Moreover date and time information might be included in the Session IDs, so excluding the date/time part, the actual length of the Session IDs might not be long enough for security purpose. Although these IDs were not increasing consecutively, a hacker might try a range of IDs so as to guess the patterns. At least that hacker had a clue to do so. For other 72% of web sites, we found that the Session IDs were either encrypted or did not contain obvious pattern.



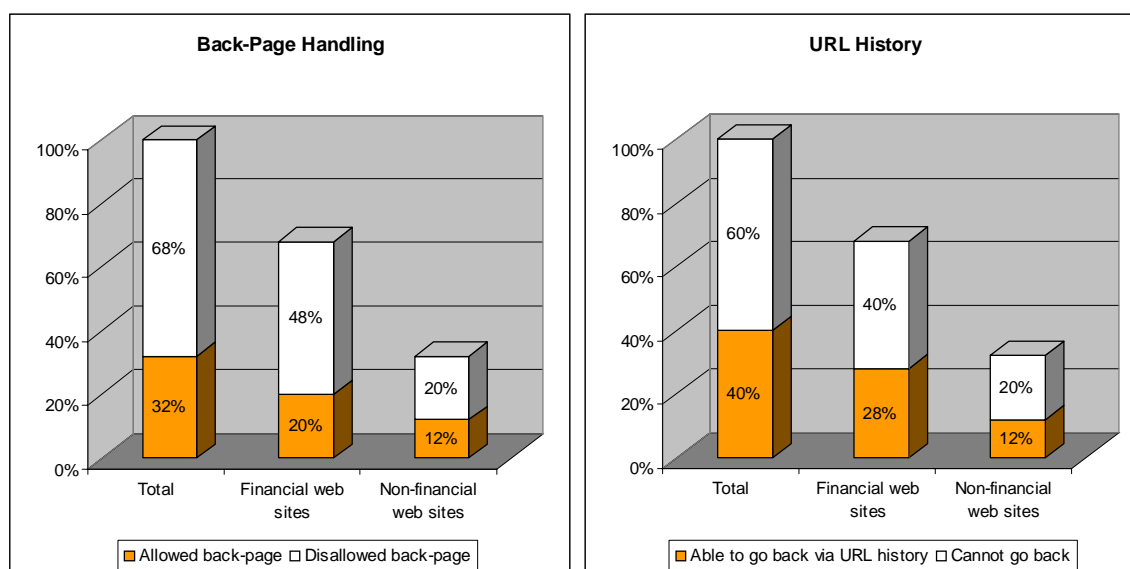
For each web site, we left the PC idle after login for 15 minutes. About 40% (32% from financial industry and 8% from non-financial industry) of studied web sites had the sessions timed out after 15 minutes.

We viewed each “Server HTTP Response Header” to check whether the web application enforced “no cache” policy. We found that about 80% (56% from financial industry and 24% from non-financial industry) of our checked web sites showed “no cache” policy were applied.



About 32% (20% from financial industry and 12% from non-financial industry) of our checked web sites allowed “back page” after logout, by clicking the “back” button of a browser. Some web sites successfully protected old information by disabling the “back” button. Also, for some secure implementation, the browser windows would be closed after logout so that the information of the closed session cannot be accessible any more.

We tried to view web pages using “history” dropdown list feature of a browser after logout. About 40% (28% from financial industry and 12% from non-financial industry) of our checked web sites were able to get access to old pages. Of course, it was a better practice not to allow history feature. In our study, we found that a financial web site used Java Applet to embed the session and thus “history feature” could not be used.

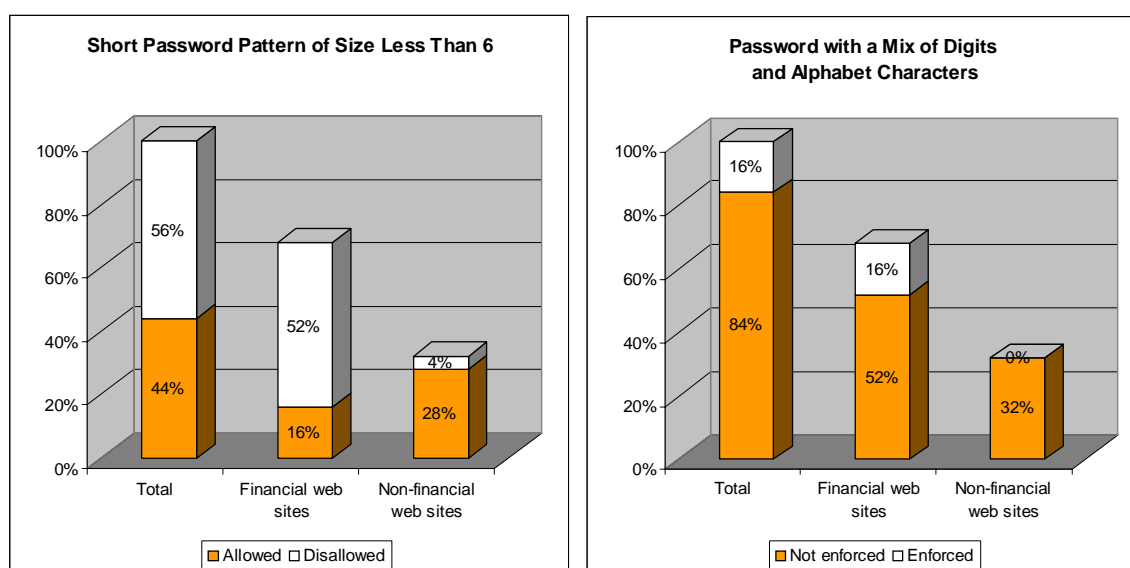


In our study, we found that about 92% (60% from financial industry and 32% from non-financial industry) of our checked web sites used cookies inside the web application. For the web sites with cookie implementation, about half of them contained sensitive information (e.g. session ID, user ID, or password). For websites with sensitive cookie information, 20% (12% from financial industry and 8% from non-financial industry) of them stored on local drive in clear text. Finally, we found that no web site made use of the “secure” parameter in HTTP to mandate encryption over cookie transmission.

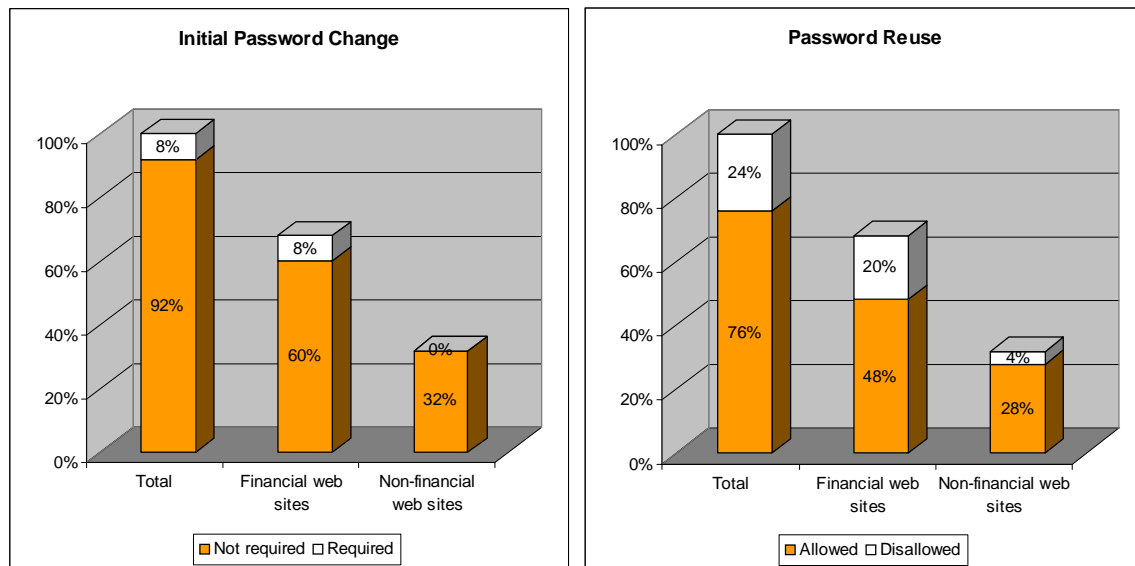
3.2.3 Password Management

“Username and Password” is a primary tool to protect sensitive data. There are 2 types of password management in web design, HTTP authentication and application level authentication. In this study, we studied the password management on the application level. We found that all the web sites rely on username-and-password as the primary mean of authentication.

Password control in different systems may be different. All of our samples were password protected. Password pattern is the key measuring factor on the secure level of password management. 56% (52% from financial industry and 4% from non-financial industry) of our samples disallowed short password patterns of less than 6 digits and/or alphabet characters. Only 16% (all from financial industry) of our samples enforced a mix of digits and alphabet characters. This may be due to the fact that some web sites integrate their web application to other systems such as a telephone application or IVR system that may disallow the input of alphabet characters. However, digit passwords (unless they are very long) do not provide a large enough combination to guard against brute-force attack.

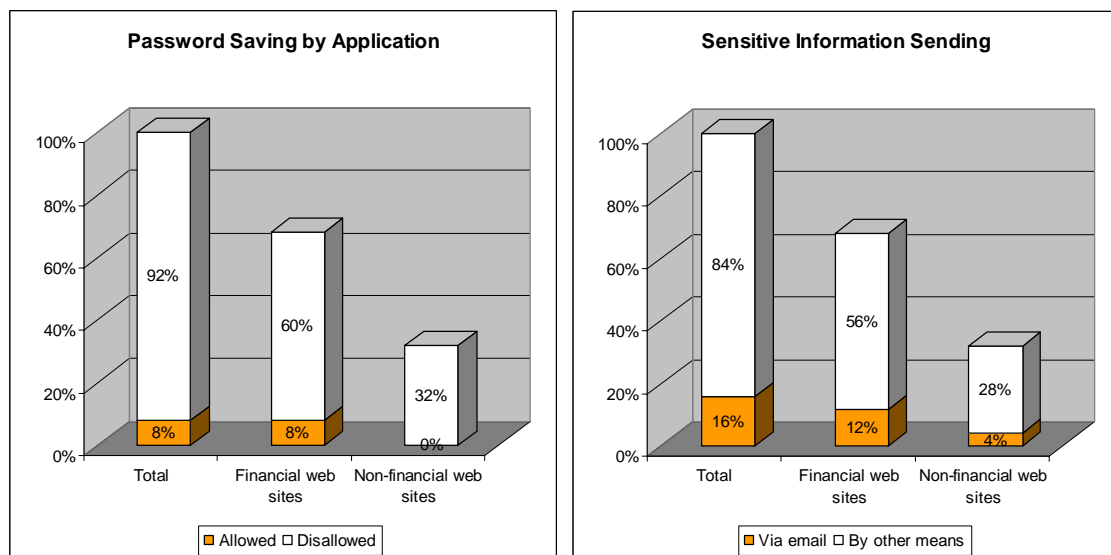


No web site imposes mandatory periodic change of password. 8% of the web sites (all from financial industry) implemented mandatory password change on first access. Please note that for web sites that allow a user to create his/her own user ID and password during registration, this requirement is not necessary. For disallowing password reuse, 24% (20% from financial industry and 4% from non-financial industry) of the web sites have this implementation.



92% (60% from financial industry and 32% from non-financial industry) of our samples did not allow saving the password in application level.

Another potential vulnerability is that some companies send out username and password information of a client account in the same email. Since email is not a secure channel, sensitive information could be intercepted or captured by hackers. We found that 16% (12% from financial industry and 4% from non-financial industry) of the web sites still sending sensitive information via emails.



Overall speaking, the results indicate that improvement on password policy is required. We think the current situation could be due to the fact that many companies may still think that strict password policy will lower usability and discourage the use of web application. Probably more promotion to the general public of the need of a stronger password policy could help improve the situation.

3.2.4 Other Exceptions

3.2.4.1 Display of unhandled database error message

During the checking, we found by occasion that a web site displayed a database error together with a database column name and a program function name during its error processing. Such database error messages may indicate that the web site could be threatened by SQL injection.

SQL injection is an exploit through injection of SQL formatting characters (e.g. single quote, comma or SQL comment characters) and appending database commands to data input fields or URL. The vulnerability might occur on any types of databases.

Since we did not intend to perform this type of checking in our checklist, so far only one case was reported.

3.2.4.2 Randomness of assignment of internal user or process ID after login

We found that a few web sites assigned an internally generated user ID or process ID after a user logged on to the web site.

Sometimes an internal generated user ID or process ID is with size of only 6 or 7 characters/digits and is just stored as plain text in a cookie or even exposed to the URL as a parameter.

If the randomness of the internally generated ID is not high enough, it would pose a security threat of the ID being hijacked for unauthorised transactions or access. In addition, if the ID is not encrypted (e.g. stored as plain text in a cookie), it would further raise the level of threat of being hijacked.

3.2.4.3 HTTP GET (rather than POST) method is used in passing login information to the web server

We found that some web sites used the HTTP GET (rather than POST) method in passing login information to the web server. Hence, the login information (e.g. user ID and password) was attached to the URL as parameters.

It could be a security vulnerability if the login information is easily exposed. The correct method should be used in this situation.

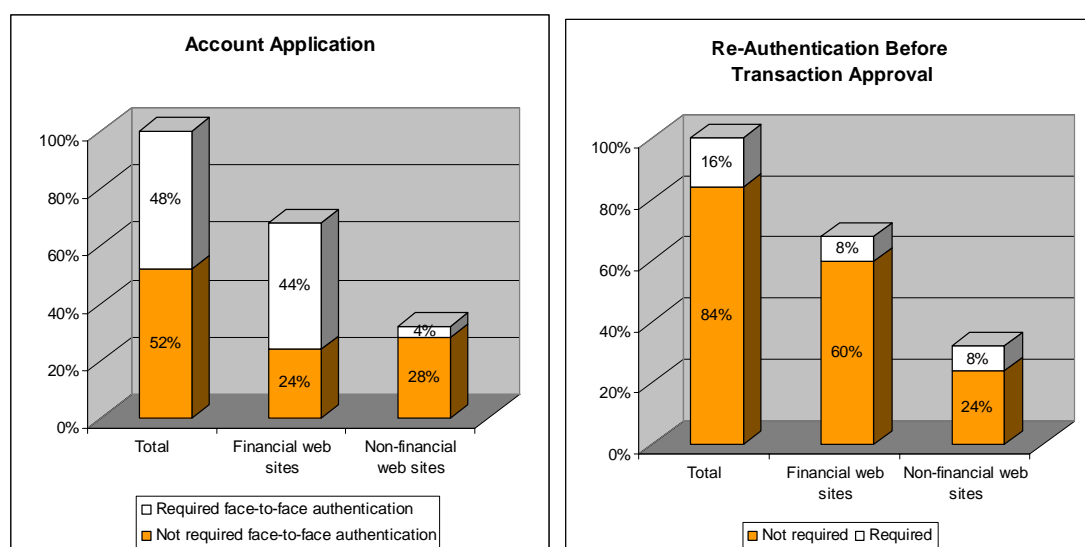
3.3 Operation Aspect

3.3.1 Control Procedure

An e-business is not just about pure technology or IT aspects. Obviously, procedures, workflow and management are integrated into the whole e-business. For secure e-business, good control procedures give complementary security to the transaction security of web sites.

In our study, we found that about 48% (44% from financial industry and 4% from non-financial industry) of our sampled web sites required a user to have a face-to-face authentication when a user applied or registered for that service. For financial industry, there is a higher need of a service provider to identify the true identification of users.

For all web sites we have checked, login procedure must be used for transaction as a basic web security implementation. Additionally, 16% (8% from financial industry and 8% from non-financial industry) of our checked web sites required a user to re-authenticate (e.g. re-enter passwords or PINs) before approving a transaction. For one of financial companies, smart card and PIN were required to authorise the transaction batch. This demonstrates additional security to users of its services.



As a conclusion of Section 3, we summarised the top ten security findings in Appendix 7.3.

4 Recommended Practices

Our recommendations are based on best practices adopted by the industry, references of reputable experts in cryptography, web security and network security.

Here we list a few major ones. “*The Ten Most Critical Web Application Security Vulnerabilities*” [OWASP] provides a very good overview of web application security and guidelines to SQL injection and buffer overflow protection. “*Analysis of the SSL 3.0 Protocol*” [Schneier], “*SSL and TLS – Designing and Building Secure Systems*” [Rescorla] and “*To Trust or Not – SSL Security Vulnerabilities*” [PISA-SSL] provide very useful guidelines on certificate management and SSL management. “*Selecting Cryptographic Key Sizes*” [Lenstra] provides a very useful guidance on the choice of cryptographic key length.

We also make reference to the “*HKSARG Interoperability Framework version 2.0*” [HKSARG-IF2] which is a technical specification on e-government for the HKSARG. Although it applies only to governmental services, it is a valuable resource to indicate the directives of the local authority as in standardising new implementations of technology which could affect business in the long run.

4.1 Infrastructure Aspect

4.1.1 DNS Zone Transfer Setting

We recommend corporations to take steps to minimise the information disclosure of corporate network topology information in DNS Zone Transfer. DNS Zone Transfer should only be allowed between authorised servers with proper authentication and if possible, in encrypted transmission.

- If a corporation uses her own DNS servers we would recommend splitting DNS services to different DNS servers for internal and external accesses. The split can minimise chances that hackers can enumerate the corporate network topology information from a mis-configured DNS server. This split also enables separation of administration duties of internal and external services to improve internal security.
- If a corporation subscribes to DNS hosting services provided by a third party, please make sure that the provider configures the hosted corporate domain so as to allow DNS zone transfer among trusted hosts only.
- We include brief steps to harden the DNS Zone Transfer of two popular DNS platforms:
 - For Microsoft DNS of Windows 2000 or 2003, by default it only allows a zone transfer to authoritative DNS servers listed in the name server. On the Properties page of Zone Transfers tab of a DNS zone, it is recommended to choose the option “Only to servers listed on the Name Servers tab” or “Only to the following servers”. If Active Directory is used, DNS information can be transmitted in a secure mode.

- For modern BIND-compliant DNS (where BIND stands for Berkeley Internet Name Domain), the “allow-transfer” directive in the file “named.conf” can be used to enforce the restriction for DNS zone transfer.

4.1.2 Web Server Software Version and Service Patch Level

Vendors release new software versions and software fixes/patches from time to time for continuous fixing of software bug and security holes as well as enhancement for new functionality.

- To minimise the security threats of the publicly known security holes/problems, regular deployment of software patches and upgrade of software versions is inevitable.
- If a corporation outsources the software support to third party vendor, it is essential for her to include in the service contract with the provision of software update and patch management services in a timely manner, so as to guarantee that any vulnerability is closed as soon as possible.
- We recognise that there is a dilemma between the need for early patching versus the need for quality management of patch that naturally prolongs the testing of patch before the production deployment, especially in mission critical services.
 - One option is to add an extra layer of application firewall [ASPECT] which usually sits on the web server or in front of it to intercept and block malicious attacks targeting at web server vulnerabilities. If this approach is taken, proper management of attack signature update is required for the application firewall.

4.1.3 X.509 Certificate Management

The trust model of current e-commerce is built on public key infrastructure where a trusted certificate authority (CA) validates the identity of the web server by signing the web server certificate with the CA’s private key. Nowadays there are usually intermediate CAs between the root CA and the server certificate. The trust model is built on the multi-level certificate chain. The ultimate source of trust of the signing CA is traced upwards to the root CA whose certificates are pre-loaded in all client browsers.

The web server certificate serves dual purposes: firstly, it authenticates the web server to the client with a properly signed server certificate; secondly, the certificate provides the public key for exchanging keys in the subsequent SSL sessions.

- We recommend that all web pages which require a client to enter sensitive information or which are used to identify a web site (e.g. containing corporate email address, contact address and PGP keys) be SSL enabled. Furthermore, these web pages should display the “padlock” on the browser so that the client can click on it to validate the server certificate (or authenticate the web server).

- Note: this merely implies enabling SSL only after submitting a login web page is not secure enough because clients cannot find the padlock to validate the certificate BEFORE inputting sensitive information.
- We recommend web servers to use SSL version which supports certificate chain (i.e. SSLv3 and TLS), and load on the server the digital certificates of the web server and all intermediate CAs in the certificate chain.
 - Note: some web servers load only the web server certificate, failing to present the client the intermediate CA certificates to provide a fully trusted certificate chain.
- Administrator should be careful to choose CA, so as to prevent certification error arising from improper CA certificate, e.g. the root CA certificate should be available on most popular web browsers.
 - Administrators are advised to choose CA with stronger key length. A reasonable key length is not less than RSA1024 [Lenstra]. In fact, RSA key length should be 1881 bits if it has to last until year 2020 and should be 2054 bits if it has to last until year 2023.
 - Administrators are also recommended to choose CA and root CA which uses X.509 v3 certificates.
- Administrator should use X.509v3 certificate which has extra security control features via extension fields:
 - Basic Constraints: use to constraint the role and position of an issuing authority or end-user certificate in a certificate chain. The basic constraints of a web server certificate should include a “Subject Type=End Entity” that restrict the certificate from issuing other certificates. As a best practice in PKI, basic constraints in end entity certificate (e.g. a web server certificate), should have a null value in the CA field while CA certificates in all levels of certificate chain should contain basic constraint field with CA = TRUE. This field should always be marked as CRITICAL [EEMA].
 - Key Usage Field: the field should state the key signing usage of certificate, e.g. digital signature, key encipherment. This field should always be marked as CRITICAL [EEMA].
 - Enhanced Key Usage: the field should state the key usage of server certificate, e.g. server authentication, client authentication. However, this field should not be marked as CRITICAL [EEMA].
 - Certificate Revocation List Distribution Point (CRLDP): the URI of the CRL distribution point should be entered [EEMA]. IE will not validate a X.509 certificate against its CRL if the CRLDP is not found in the certificate [PISA-SSL].
- Administrators are reminded to use the digital certificate of the same correct domain in the web site.
 - Note: some corporations have several domain names which they use interchangeably, e.g. company-a.com and company-a.com.hk. We warn against the use of digital certificate of another domain in the web site. Digital certificate of another domain will generate alert messages and confuse the users of the trustworthiness of the web page.

- In case of relocation of a web site to a different domain name, a new certificate that matches the new domain name is necessary for the new web site.
- Sensitive information should not be transmitted to another non-secure domain or URL. All sensitive data should be protected by SSL.
 - Note: We warn against the practice of incorporate non-HTTPS form submission in HTTPS pages.
- Administrators should ensure their SSL certificate be valid for service all the time. Administrator should renew their SSL certificate before it expired (certificate lifetime is usually 1-2 years). Expired or invalid SSL should not be use in production environment.

4.1.4 Server SSL Configuration

SSL is the flagship transport protocol used in e-commerce nowadays. It must be carefully configured to be secure.

- The choice of SSL version determines the security that the infrastructure can attain. Do not configure the web site to use SSLv2 as primary SSL protocol as it has many design flaws [Schneier]. Configuring the web site to be downward compatible to SSLv2 is also not recommended as SSLv3 and TLS 1.0 (which is regarded as SSLv3.1) are already mature and widely supported.
- TLS 1.0 provides the best security as the transport protocol. We agree with recommendation put forward in the Interoperability Framework of the HKSAR government for e-government services, “*New implementations should ready themselves to support TLS and should ensure their TLS implementation’s backward compatibility with SSLv3 where situation allows*” [HKSARG-IF2]. This advice is also applicable to business use of SSL.
- Administrators should be aware that SSLv3 and TLS are so flexible in the choice of ciphers (suites of authentication, encryption and MAC algorithms) and a negligent configuration could render the web server extremely insecure.
 - Administrators should NOT choose “any” in ciphers which would include a number of insecure ciphers.
 - One good example is SSLv3 and TLS can support *no encryption* with RSA authentication and a MAC.
 - Administrators should not choose low grade ciphers like DES-56, RC4-56, RC4-40, RC2-40. They can be cracked in reasonable time. Moore’s law states that computing power doubles every 18 months. We could expect the life span of low grade ciphers not able to survive reasonable period of usage. For example, DES was cracked in 4 days in 1998. [Lenstra]
 - Administrators should enable web server to use 3DES, AES and RC4-128 for encryption. If for performance concern, RC4-128 can be the default encryption algorithm. SHA-1 should be selected as the MAC algorithm. Here are some sources of support:

- [Rescorla] proposed to use 768-bit key RSA or DH/DSS; use 3DES for better security or RC4-128 for better performance; and use SHA-1 for integrity check
- [HKSARG-IF2] recommended DES, 3DES and AES for encryption and SHA-1 for MAC. PISA agrees to the recommendation except for DES which is no longer regarded secure enough.

4.2 Application Aspect

4.2.1 Web Page Management

The objectives of web page management are to ensure the confidentiality and integrity of sensitive information on the web server and ensure web requests are authenticated.

To protect the sensitive data presented on the page, the information should be transmitted via an encrypted (say, SSL) channel. Furthermore, never trust client side data [Brassinne] – do not store critical information on the client and do not only validate data at client side.

Besides sensitive data & data fields on the page, the program structure and logic should be protected as well. Before web pages are pushed to the production server, they should be filtered to strip off comments (“honest” disclosure) and avoid using hidden data fields for sensitive data items. Program logics should also reside on the server instead of client side as far as possible. [OWASP]

To ensure that a web page request is originated from a page generated by the web server or a trusted web server, some web sites check the 'referer' field. [Brassinne]

4.2.2 Session Management

If a session token is captured in transit through network interception, a web application account is then trivially prone to a replay or hijacking attack. Typical web encryption technologies include but are not limited to SSL and TLS protocols in order to safeguard the state mechanism token.

For session management, we shall further discuss in terms of (1) system time leakage, (2) session ID sequence, (3) session time-out, (4) “no-cache” policy, (5) “back” page allowance, (6) “history” feature after logout, and (7) Cookie.

System Time Leakage

It may be unavoidable to show the system time of the server as a proof of the time of transactions. If session ID or cookies should be used and they are composed of the time element, they should be long enough to reduce predictability.

Session ID sequence

Session ID should be generated in an un-predictable sequence. In January 2001, Netscraft released an advisory related to the session IDs generated by some version 2.0 of JSDK for the Java Web Server, IBM Websphere and ATG Dynamo e-Business platforms [Netscraft]. Netscraft identified the simple manner in which session IDs were encoded. In addition to be generated in unpredictable patterns, session ID should be long enough to reduce the chance of brute-forced attack.

Session Time-out

Session tokens should be time-out properly. Theoretically, the length of the time-out should be slightly longer than the expected longest session time of a transaction. Session tokens that do not expire on HTTP server can allow an attacker unlimited time to guess or brute force a valid authenticated session token. Time-out can protect the application from session replay attacks.

No-cache policy

After time-out or normal session quit, it should not allow old information, or old pages, to be viewed by other browser users. Several steps can be taken to help prevent sensitive data being accessible to those hackers or non-intended user. In general web applications, a system designer should assume all users will connect from a public terminal. To disallow the access of old pages, the following HTTP header settings can be used:

- “Expires: -1”
 - To ensure that a page has already expired immediately and a browser will always get a fresh copy
- “Pragma: no-cache”
 - Whilst preventing proxy servers from caching content served through them is extremely difficult, the pragma cache meta tag tells proxy servers not to cache pages with the tag set to no-cache. This parameter is applicable to HTTP 1.0 only.
- “Cache-Control: no-cache, must-revalidate”
 - No-cache tags tell browsers not to cache pages and must-revalidate requires re-validation with server in accessing expired pages. A side effect is that the “back” page button will typically send a pop-up asking the user to repost the request.

The above HTTP header settings can be done on web site level (i.e. for all web pages under a web site), web application level (i.e. for all web pages of an application hosted in an application server) or managed individually in each web page. The choice is dependent on the requirement of the web application.

Back-page Allowance

Many web applications hide or disable the “back page” button of the browser to disallow a user to review information on the previous page. While this is a good practice, some application forgot to disable the equivalent hotkeys (e.g. Alt-<left-arrow>).

Browser Histories

Browser histories contain URL history information including extended path data for all sites visited. URL history lists may contain sensitive data that could be a result of using HTTP GET request instead of POST in submitting the form. Also, if page expiry setting is not enabled, old pages that contain sensitive information could be accessible using the

URL history. Hence, the web application should be designed and configured properly in order to disallow improper access via URL histories.

Cookies

It is suggested to avoid storing sensitive information in cookies in the first place. If it is unavoidable to do so, the cookie should be encrypted before storing to the local hard disk.

4.2.3 Password Management

Many current systems employ password or PIN as authentication and access control mechanism. Password protection is thus the most critical aspect to protect information.

Password composition is the key factor of the password security. We recommend web application password be at least 6 characters long and a combinations of alphabet, digit and symbols. Dictionary words should not be accepted as password.

During the first login, users should be forced to change the given password immediately and certain mechanism be employed to ban password reuse for a certain period of time. It is also recommended to enforce user to change password periodically. As an operational control, users should be educated to use different password for critical services.

At system level, we recommend banning storage of password information in cookies or cache. Such practice merely let unauthorised user get through the login process. Web application should not provide password saving function to users.

On the distribution of username and password, we recommend sending them out separately using different channels, e.g. mail or face-to-face. Because email is not a secure channel, whenever possible, it is better not to use this channel to send sensitive data.

4.2.4 Others

4.2.4.1 Reducing the threat of SQL injection

[Note: Because of legal and ethical reasons, PISA did not conduct SQL injection test which was intrusive to the web sites. However, we include recommendation here for completeness.]

SQL injection is a kind of database attack. To reduce the possibility of the exploit, the following is recommended in the design and implementation of a web site:

- Perform item content checking/validation properly before creating or formatting SQL statements for database query or update.
- Include proper error handling in the programs to avoid showing unhandled error messages to the client browser. Instead, unhandled error messages could be written to application log files for troubleshooting by the web site.
- Configure to run web application using database accounts with the least required database privileges

4.2.4.2 Reducing the threat of Buffer Overflow attack

[Note: Because of legal and ethical reasons, PISA did not conduct Buffer Overflow penetration test which was intrusive to the web sites. However, we include recommendation here for completeness.]

A buffer overflow occurs when the size of data received from the client is larger than the size of the buffer allocated to store the data. Data is then copied outside the border of the buffer. Attackers can craft client data such that the extra data may contain malicious code or may crash the system.

We recommend checking boundary for all data input or use program languages which automatically checks bounds of buffers. Furthermore, library modules which implements safe, bounds-check buffers can be used.

To minimise the risk of system being exploited through buffer overflow vulnerabilities, it is always a good practice to run the program using unprivileged account different from the web server service account [PISA-BO].

4.2.4.3 Improve the randomness of internally assigned user ID or process ID

If a web site would like to use internally generated user ID or process ID for further processing after a user logged on to the web site, it is important to use algorithm that could generate IDs of high randomness. In this way, it makes the IDs less predictable and could reduce the chance of being hijacked.

Furthermore, if the IDs are kept in cookies, they should be encrypted instead of just in plain text.

4.2.4.4 Use HTTP POST (instead of GET) method in passing sensitive information over the web

HTTP POST method should be used when posting sensitive information (e.g. user ID and password) to the web server. Otherwise, if HTTP GET is used, the sensitive information will be attached to the URL as parameters and are visible to the end-users.

For instance, a password reset application uses a GET request. When a user enters the information, the HTTP request would actually look like this:

<http://www.victim.com/cgi-bin/password?username=testuser&newpassword=testpasswd>

Common data disclosure include credit card numbers, account numbers, passwords and private data like addresses. If an application uses the POST method, the parameters are sent as the entity body of the request and therefore it was not visible. It is therefore necessary to use POST to submit forms.

4.2.4.5 Only trust active content of trusted web sites

During the checking, we found the process of transactions of some web sites is wrapped in an applet. It is quite popular since the use active content (e.g. Javascript, Java applet or ActiveX control) on the client browser can improve the user-friendliness in using the webpage and on the other hand, could improve the overall performance by offloading some of the server's workload.

The idea of active content makes use of small executables or script code that could be executed in a browser in order to provide dynamic and rich user-interface experience. Also, it could offload the workload of the server since part of the processing logic is handled by the small executables or scripts that run on the client browser.

To avoid active content attack (e.g. via ActiveX control), it is recommended to only trust the active content download objects/controls that are signed by trusted web sites.

4.2.4.6 Error handling

Improper handling of errors can introduce a variety of security problems for a web site [OWASP].

PISA recommends careful coding of error handling routines using the "deny access until specifically granted" mechanism. The opposite mechanism, namely "grant access until denied", is erroneous.

Overly informative error messages can give attacker helpful information to develop the attack strategy, e.g. different reject codes on the login page as "non-existing userid" and "wrong password" error could provide feedback information to hacker to increase the efficiency of the attack. We recommend minimise disclosure of overly detailed error messages to the users.

4.3 Operation Aspect

4.3.1 Control Procedure

Even if an e-commerce web site deploys the first class Internet transaction security technologies in the world, a hacker can still compromise a system using social engineering techniques to attack human, the weakest link. So we also need operational controls to contribute to the overall e-commerce security design. The holistic security design should cover how a user interfaces the business at a very first moment, how a user registers to the service, how a user uses that service, how the service is operated, how information is transmitted and stored, how a service provider handles the relation with the users, etc.

We recommend that user account opening should require face-to-face authentication for highly sensitive e-business services. Although there is considerable overhead in face-to-face authentication, it can be justified by the quality and security achieved.

In today's common practice, a transaction cannot be made without login. For highly secure system, we recommend that user re-authentication be enforced. With such protection, a hacker gaining access to the session ID cannot crack the system unless he/she also possess the user ID and password. Moreover, re-authentication also provides a form of non-repudiation.

5 Future Challenges

The complexity of web security goes in parallel with the developing web technologies to adopt. Nowadays, there are abundant choices of components (say, hardware platforms, operating systems, standards, protocols, program languages and software tools) to build up one's web site. To cite a few of them, we have:

- multiple protocols/standards – e.g. HTTP, HTTPS, SOAP, X.509, LDAP, etc.
- multiple programming language/tools – J2EE, .NET, WML, SQL, XML, PHP, PERL, Message Queues, etc.
- multiple software/hardware platforms – e.g. web and application servers run on Microsoft Windows platform on server PCs but databases run Unix-based servers
- multiple tier application architecture – e.g. the client tier could be on .NET platform while the business-logic tier could be on J2EE platform
- complex networking infrastructure – firewalls, IDS, LAN/WAN, etc.
- demanding system requirements – high availability, disaster recovery, mirroring sites, etc.

This is not yet an exhaustive list so we can imagine the challenges to manage the web security. Furthermore, with the evolution of web services and greater use of XML, we are going to face a new era of web security management. XML statements can embed binary content (e.g. a picture in JPG format, a X.509 security certificate or a program executable) in Base64 format and transmit via HTTP protocol. Web services may then extract the binary content for further processing or execution. Since all these are possible just via the HTTP protocol, it will render traditional firewalls, content filtering systems or IDS useless in protecting web services and XML security. Researches on XML infrastructure security are underway to address these issues [PISA-XML].

To cope with the ever changing technologies and manage web security successfully we need a team work to contribute to specialisation in various key areas. At the same time, we need a general understanding of the overall architecture. It would be a big challenge and thus an opportunity for all proficient security professionals.

6 Reference

[ASPECT] *Application Security: a Comparison of Approaches*, Aspect Security Inc., 2003

<http://www.aspectsecurity.com/owasp.html>

[Brassinne] Pierre de la Brassinne, *Web Application Security for Managers*, SANS Institute, August 24, 2002

<http://www.sans.org/rr/paper.php?id=27>

[EEMA] *The PKI Challenge Best Practice for PKI Users*, EEMA, April 2003

<https://www.eema.org/pki-challenge/latest.asp>

[HKSARG-IF2] *The HKSARG Interoperability Framework version 2.0*, HKSARG, Nov. 2003

http://www.itsd.gov.hk/itsd/english/infra/eif.htm#if_spec

[IBM] *HTTP Server: Migration*, IBM Corporation, 30 October 2003

<http://www-1.ibm.com/servers/eserver/series/software/http/product/migrate.html>

[Lenstra] Arjen K. Lenstra, Eric R. Verheul, *Selecting Cryptographic Key Sizes*, Nov 15, 1999.

<http://www.cacr.math.uwaterloo.ca/conferences/1999/ecc99/lenstra.doc>

[Microsoft] *Frequently Asked Questions About Availability and Support – “When did Microsoft first announce the retirement of the Microsoft Windows NT Server 4.0 operating system?”*, Microsoft Corporation, 29-Jan-2003

<http://www.microsoft.com/ntserver/productinfo/availability/faq.asp?bPrint=True#1>

[Netcraft] *Security Advisory 2001-01.1 - Predictable Session IDs*, Netcraft, Jan 2001

http://news.netcraft.com/archives/2003/01/01/security_advisory_2001011_predictable_session_ids.html

[Netscape] *Netscape Enterprise Server 3.6 – End of Life Statement*, Netscape, 25 April 2000

<http://help.netscape.com/products/eol/server/es36-eol.html>

[NP] *UK's Internet Infrastructure Open to Prying Eyes – DNS Zone Transfers Allowed from First and Second Level Domains*, Network Penetration, 2003

<http://www.networkpenetration.com/ukdns.html>

[OWASP] *The Ten Most Critical Web Application Security Vulnerabilities*, Open Web Application Security Project, January 27, 2004

<http://www.owasp.org/documentation/topten>

[PISA-BO] *Buffer Overflow & Writing Secure Code*, PISA Interactive Workshop Presentation, Jun 2002

<http://www.pisa.org.hk/event/bo.pdf>

[PISA-SSL] *To Trust or Not – SSL Security Vulnerabilities*, PISA Interactive Workshop Presentation, Nov 2002

<http://www.pisa.org.hk/event/ssl1102.zip>

[PISA-XML] *XML Security*, PISA Interactive Workshop Presentation, July 2003.

<http://www.pisa.org.hk/event/xml0726.htm>

[Rescorla] Eric Rescorla, *SSL and TLS – Designing and Building Secure Systems*, Addison Wesley 2002.

[RFC3280] Network Working Group — R. Housley, *RFC 3280 – Internet X.509 Public Key Infrastructure*, April 2002.

<http://www.ietf.org/rfc/rfc3280.txt>

[Schneier] David Wagner, Bruce Schneier, *Analysis of the SSL 3.0 Protocol*,

<http://www.counterpane.com/ssl-revised.pdf>

7 Appendix

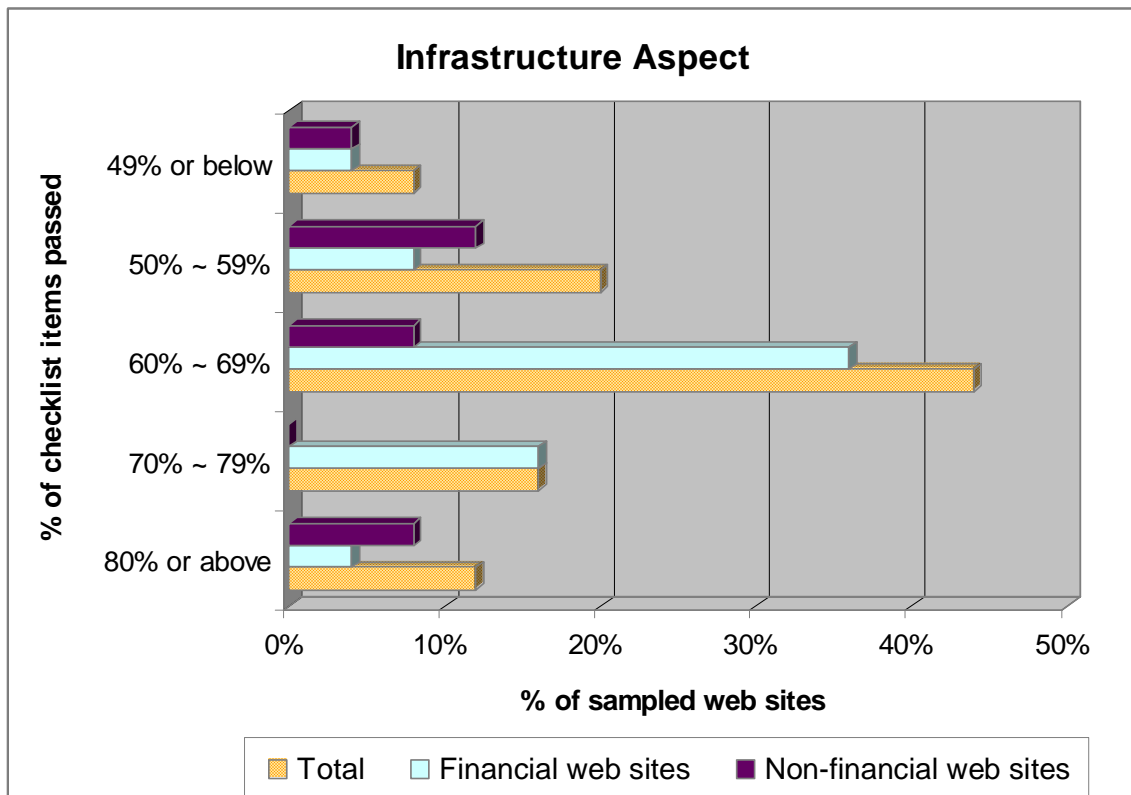
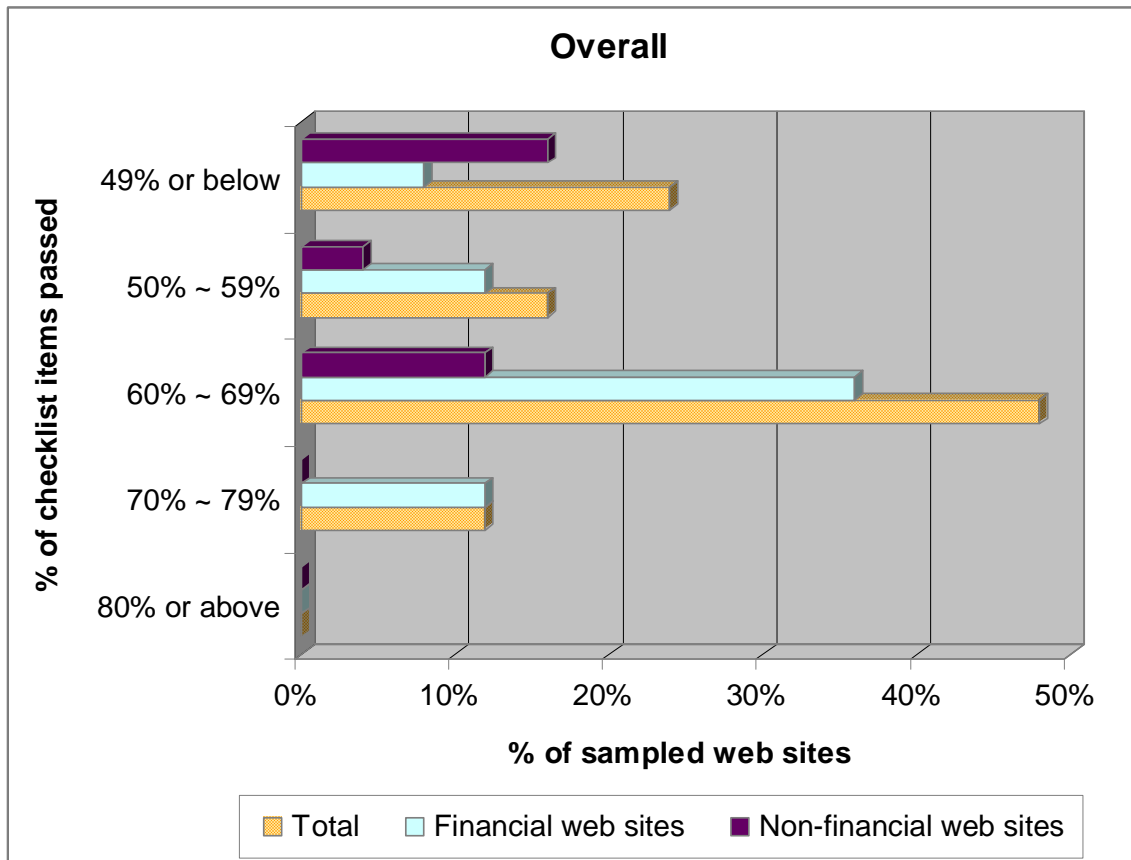
7.1 Hong Kong E-Commerce Security 2003 Checklist

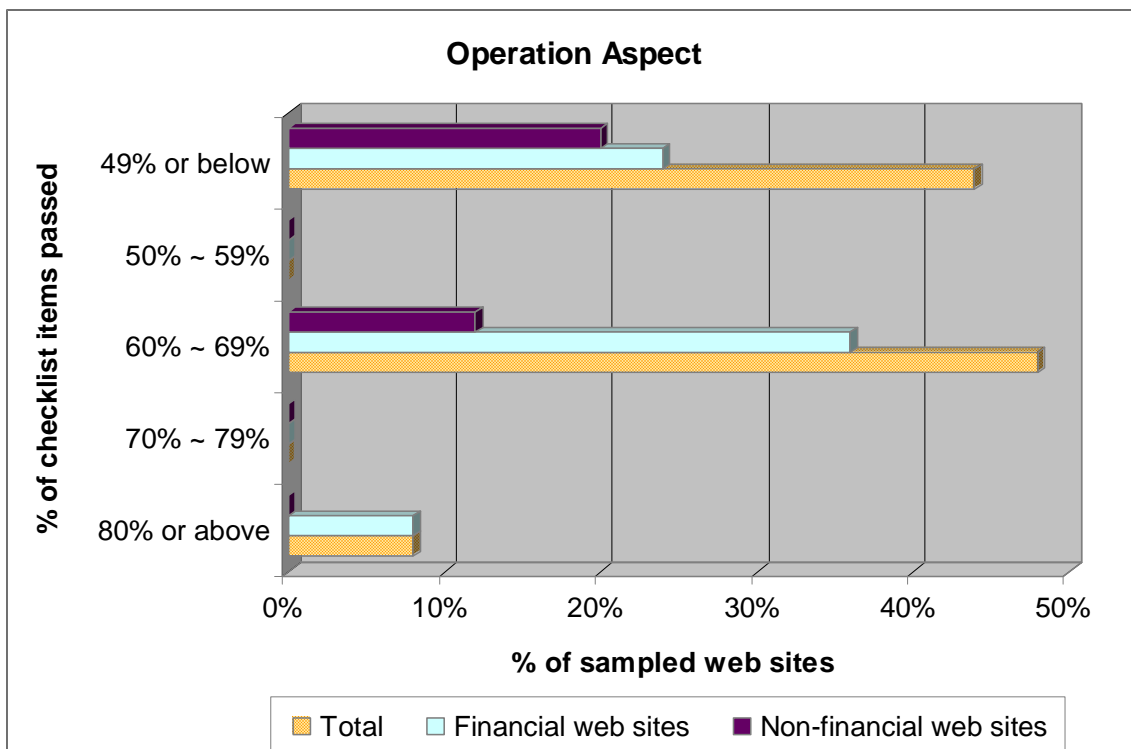
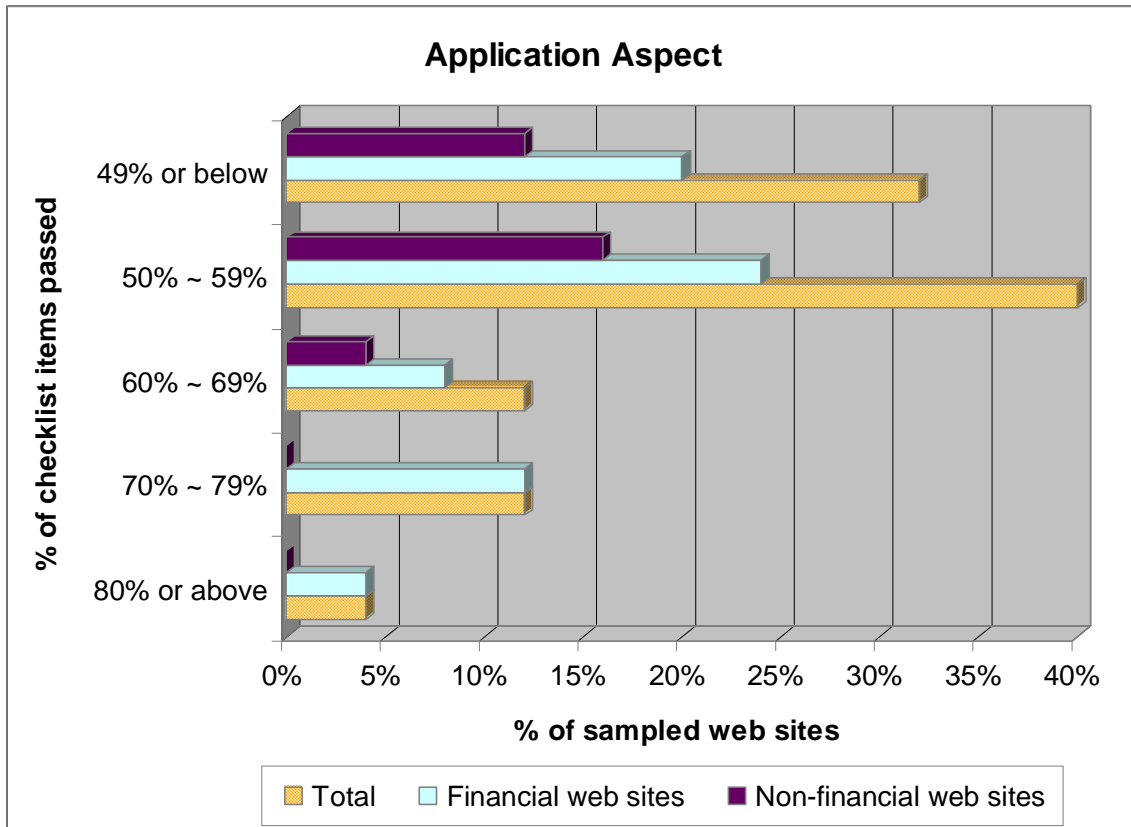
Areas of Review	Check Item	Methods/Tools (expected answers)
1. Infrastructure Management	1.1 Is the DNS infrastructure of the web site protected from DNS zone transfer to stop information leakage?	(a) Find the target company's dns server (b) Use zone transfer command to get info from the company's dns server and see if successful (c) Query Refused? (yes, no)
	1.2 What software is the web server using? Is it using the latest version and service pack?	Check the HTTP response from server - Write down info of (SW name, version, service pack info)
2. Web Page Management	2.1 Do the HTML pages contain sensitive information?	Use browser's view source function to read the HTML code - Look for type=hidden, name=Bypass, Login, Passwd or check for autocomplete=off (this may be a password field) (yes, no)
3. Session Management	3.1 Does the server leak real system time? - server respond with running system clock time	Check the HTTP response from server for system time several times. Does it leak the real system time? (yes, no, not_sure)
	3.2 Is the session ID generated with a predictable sequence? - e.g. simple increment	Check the HTTP response from server and look for session ID such as ASPSESSIONID, SESSIONID, JSESSIONID, SID, session_id. - Record the Session ID for 5 times. What is the total length of Session ID? - Note the changing pattern. What is the effective length of Session ID after taking out date/time pattern? - Is the session ID in a predictable sequence? (yes, no)
	3.3 Will the session be time-out?	Leave the PC idle after login for 15 minutes. Is the session timed out? (yes, no)
	3.4 Does the web application enforce "no cache" policy?	Look for the following in Server HTTP Response Header Pragma=no_cache, Cache-Control=no cache (yes, no)
	3.5 Is the back page allowed after logout ?	By clicking the "back" button. (yes, no)
	3.6 Can web pages be viewed using "history" features after logout ?	By clicking the "history" dropdown list to access the old pages after logout . (yes, no)
	3.7 Does the web application use cookie? Is the cookie used in a secure way?	3.7.1 Use Netscape - Does web application use cookie? (yes, no, not_sure)
3.7.2 Does cookie contain sensitive information, such as session ID, User ID and/or password? - Check the cookie content using Netscape (yes, no, not_sure)		
If no, skip to 4. 3.7.3 Is the cookie that contains sensitive information stored on local drive in clear text? - Check if "expires" parameter is set in the "Set-Cookie" of the HTTP response header and with value > current date? - If yes, check if the cookie content is in plain text? (yes, no, not_sure)		

Areas of Review	Check Item	Methods/Tools (expected answers)
		3.7.4 Is the cookie that contains sensitive information sent over HTTP w/o SSL? - Check if "Secure" parameter is set in the "Set-Cookie" of the HTTP response header ? (yes, no, not_sure)
4. Password Management	4.1 Does the system require both a username and password?	By observation (yes, no)
	4.2 Does the system encourage a strong password policy?	4.2.1 Min length = 6 or above? (yes, no)
		4.2.2 Mix of digit and alphabet chars (force_digit-only, force_mix, not_force_any, others)
		4.2.3 Force periodic change of password? (yes, no)
		4.2.4 Force a change on first access? (yes, no, n/a)
		4.2.5 Disallow password reuse? (yes, no, n/a)
4.3 Does the web application provide password saving function?	Is there an option to allow saving password? (yes, no)	
4.4 Does the web site send out user ID and password together after registration?	By user experience (yes, no, n/a)	
5. X.509 Certificate Management	5.1 Is SSL turned on during login?	Check that SSL is activated on or before login. (warning is given when entering encrypted page) (yes, no)
	5.2 Is encrypted session clearly indicated on the browser?	Check that a padlock is displayed on the browser during HTTPS session. (yes, no)
	5.3 Does the server provide the cert chain to the browser when in SSLv3 or above	Use Netscape. Check that the Intermediate-CA cert is provided (yes, no, no_sub_CA)
	5.4 Is the server SSL X.509 cert of version 1 which is not secure?	Check the X.509 version by double-clicking the padlock icon of the browser (v1, v3)
	5.5 Is the issuing CA of the SSL cert a HK SARG recognised CA?	Follow the Certification Path Tab page - Name of CA - Recognised CA (yes, no)
	5.6 Is the issuing CA's X.509 cert of version 1 which is not secure?	View the Issuing CA cert's version field (v1, v3)
	5.7 Is the SSL server certificate within valid period?	View "Valid to" field to see if expired (valid, expired (expiry_date=...))
	5.8 Is the SSL server certificate chain valid?	Check if certification path is complete without error (yes, no)
	5.9 Does Basic Constraints key field exist? e.g. "End Entity" for server cert, "CA" for issuing CA	check server and CA cert details - Root-CA (yes, no) - Int-CA (yes, no) - Server (yes, no)
	5.10 CRL distribution point exists <u>for all certs</u> ?	check server and CA cert details - Root-CA (yes, no) - Int-CA (yes, no) - Server (yes, no)
	5.11 Server Cert Enhanced Key Usage exists <u>for all certs</u> ?	check server and CA cert details - Root-CA (yes, no) - Int-CA (yes, no) - Server (yes, no)
	5.12 Does the Issuing CA "Key Usage field" exists and defined? (e.g. Digital Signature, Non-Repudiation, Key Encipherment, Data Encipherment, Certificate Signing)	check server and CA cert details - Root-CA (yes, no) - Int-CA (yes, no) - Server (yes, no)

Areas of Review	Check Item	Methods/Tools (expected answers)
	5.13 Is the CA and server public key strong enough?	Record public key length and algorithm of CA and Int-CA - Root-CA (RSA1000/RSA1024/RSA2048) - Int-CA (RSA1000/RSA1024/RSA2048) - Server (RSA1000/RSA1024/RSA2048)
6. Server SSL Config	6.1 What is the default ciphersuite and encryption key length of web site?	Use Netscape Navigator with all SSL version and ciphersuite enabled. Turn on all warnings in " Preference Privacy&Security SSL " Connect and double-clicking the padlock. Check the security tab of the Page Info. - Record the encryption used (e.g. RC4-128, RC4-56, 3DES-EBC-CBC-168)
	6.2 Is the site running a secure version of SSL (i.e. SSLv3 or TLS1.0)?	6.2.1 Use Netscape Navigator to control client's SSL version - Can client use SSLv2 only to connect? (yes, no) 6.2.2 Can client use TLSv1 only to connect? (yes, no)
	6.3 Does the site allow "no encryption but MAC only" ciphersuite?	Use Netscape Navigator - select SSL 3.0 & " no encryption with an MD5 MAC " ciphersuite and try to connect - Can you connect? (yes, no)
	6.4 Does the site allow strongest ciphersuite (3DES-CBC-SHA1)?	Use Netscape Navigator - select SSL 3.0 & " 3DES and a 168 bit key and a SHA-1 MAC " ciphersuite and try to connect - Can you connect? (yes, no)
	6.5 Does the site support low grade encryption with 40/56 bit key?	Use Netscape Navigator, select SSL3.0 & enable only these ciphersuite one by one <u>DES-56, RC4-56, RC4-40 and RC2-40.</u> - Can you connect with these? DES-56 (yes,no) RC4-56 (yes,no) RC4-40 (yes,no) RC2-40 (yes,no)
	6.6 Does the web site redirect form submission to another domain?	Use Netscape Navigator to browse the SSL form page. - Check the domain of the form page and the originated domain. - Are the domains the same? (yes, no)
	6.7 Does the web site redirect form submission to non-SSL URL?	Use Netscape Navigator to browse the SSL form page. - View " Page Info Forms " to locate the form submission address" - Is the form submitted to non-SSL URL? (yes, no, not sure)
7. Operational / procedural control	7.1 Does account opening require face-to-face authentication?	By user experience (yes, no)
	7.2 Can a transaction be made without login?	By observation (yes, no)
	7.3 Apart from the initial login verification, are they other authorization control (e.g. re-entering PIN) before approving a transaction?	By observation (yes, no)
8. Other exceptions recorded	8.1 Name of exception and detail description	e.g. - Int-CA cert with different serial #, - Error page leaks database structure, etc.

7.2 Survey Statistics





7.3 Top 10 Security Issues

Top 10 Security Issues
<ul style="list-style-type: none">• Web server software version not new enough• No time-out of login after idle for 15 minutes• Not enforce a mix of digits and alphabet characters for passwords• Not mandate periodic change of password• Allow password reuse• Key Usage not defined for all levels of certificates• CRL distribution point not included for all levels of certificates• Still support SSL v2• Still support low grade encryption algorithms in SSL connection• Allow no encryption in SSL v3 / TLS

The End