Australian Communications-Electronic Security Instruction 33 (ACSI 33)

Point of Contact: Customer Services Team

Phone: 02 6265 0197 Email: assist@dsd.gov.au

HANDBOOK 10

WEB SECURITY

Version 1.0

Objectives

1001. Web security mechanisms on information systems may have some or all of the objectives listed below. This handbook focuses on the functionality and requirements for web security controls.

a. To protect the integrity of information submitted to, contained within or retrieved from the web site

b. To protect the confidentiality of identified information by restricting access on a need-to-know basis

c. To protect the availability of the system by controlling access to critical system functions

Web Security Issues and Terminology

1002. The world wide web is an intuitive, user friendly, easily navigable system that has become an essential communication tool for government.

1003. Some of the common terms that are used in this handbook follow:

a. HTTP is the Hypertext Transfer Protocol. HTTP is a protocol used to transfer information from the web server to the web browser over the network.

b. A web server is a software tool that allows web browsers to connect over the network, and retrieve specific information using the HTTP protocol.

c. A web browser is a software tool run by an end-user that is able to connect to a remote web server, retrieve information via HTTP, and display information that is presented in 'HTML' or 'XML'.

d. HTML is the Hypertext Markup Language. HTML is an evolving standard for creating web pages.

e. XML is the 'Extended Markup Language', and the emerging standard for creating rich content, flexible web-based documents and application interfaces. XHTML is the bridge standard between HTML and XML.

f. A URL is a Universal Resource Locator. A full web page reference, which includes the site at which the page resides, the directory in which can be found, and the name of the file in question makes up a URL.

g. A 'cookie' is a small file containing information that is uploaded to the browser by some sites. The site can then query the file when the site is next visited.

1004. Installation of web server technology creates a 'window' into an agency's network that can potentially be misused by attackers. A poorly configured or maintained web server is likely to introduce problems that allow unauthorised remote users to perform actions outside the scope of legitimate activity, impacting on confidentiality, integrity or availability. Examples of such actions are:

a. Users examining files on a web server that would not normally be released via the web.

b. Retrieve information about the server computer, which may allow a potential attacker to target the computer more effectively.

c. Execute commands remotely on a server computer, which may allow the attacker to change the system or web pages in some way.

1005. Personal privacy when using a web browser is not guaranteed. Web browsers leave the digital equivalent of footprints on each site that is visited. These 'audit footprints' can be used to identify the web browsing habits of an individual or the organisation in general. Privacy is further voided when 'cookies' are enabled in the client browser. Cookies are small files that can be 'attached' to a browser by visited sites. On return, the site or network that uploaded the cookie can retrieve or modify the information.

1006. Inappropriately configured client software may allow remote sites or users to perform actions on the client computer such as: access files on the local computer, introduce computer viruses, steal processor time from the local computer, crash the browser software, or crash the host computer. Software downloaded from remote sites and executed locally may contain malicious code

that impacts negatively on the client system.

1007. Unless appropriate encryption technology is in use, information that is sent over a network can be intercepted and analysed at any point in the many network hops that are likely to be between the client computer and the remote web server.

Anonymity and Privacy

1008. Each time a web browser is used to view a web page, the digital equivalent of footprints are left at each site. The web server at each site keeps details of:

a. The IP address from which the page request originated.

b. The user's name (if the user has been asked for authentication by the site, or the browser's machine is running an identification utility known as 'identd').

c. The URL (Universal Resource Locator - usually a web page) that was accessed on the site.

d. The status of the request.

e. The amount of information that was transmitted back for this page.

1009. Some browsers also provide additional information such as: the operating system browser's host computer, the type of browser in use, the version of the browser, the user's email address, and the last web page that was visited.

1010. Although any one web site only receives log entries from pages that it actually hosts, the 'referrer' log contains a pointer to the previous site that was visited. This pointer can be quite valuable to the owners of some sites. An example is available in **Annex A**. The web browser's ability to disclose details to host sites can lead to embarrassing situations such as a breach of the organisation's acceptable use policy. Consider the following situation:

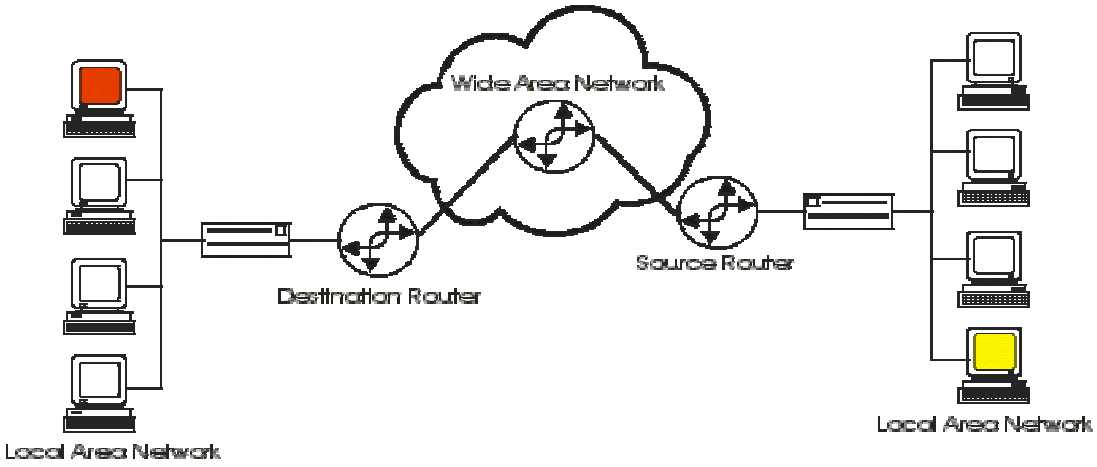
You are a regular visitor to a site that contains useful work-related information. The site recently added an article on management techniques, and illustrated a point with a cartoon that was sourced from dedicated cartoon publishing site. The cartoon site retrieves your email address from the web server logs, and automatically adds you to the 'cartoon' mailing list - which sends you one cartoon a day. The images in question breach your site's acceptable work-related computer use policy.

The cartoon site sells its mailing list to a direct email marketing company, which then sends you several hundred messages a week, impacting on your mail server availability.

1011. Software exists that can remove some of the fields that a browser would normally give out. This capability is normally built into a web proxy server that forms part of a firewall.

Data Confidentiality

1012. Modern networks are structured in such a way that information usually passes through several network 'hops' to get from source to destination. In the following diagram, if a user from a machine on the right wishes to access a web page from a machine on the left, there are several places where the communication can be intercepted.



1013. The communication between source and destination can be potentially intercepted by:

a. **Any user on the source local area network.** A local area network usually operates in 'broadcast' mode. Each station 'shouts' over the network so that the destination host, or any network devices that create a path to the network host, can 'hear' it.

b. **The administrator of the source router.** The source router is responsible for creating a network path between the source machine and the destination machine, and as such carries the communication.

c. **The administrators of any intermediate network devices such as routers or firewalls.** On an intranet, some level of trust can be safely assigned to the network devices, and those staff that are performing administration tasks. The Internet, however, is a dynamic system that will re-route communications in response to degraded traffic flow or service interruption. The source machine has very little control over which path communications will flow, and as such, cannot guarantee the integrity of

administrators at each network device along the path from source to destination.

1014. Encryption technology is clearly the most effective mechanism to provide data confidentiality between source and destination. Encryption of web services is discussed later in this handbook (including references to other handbooks).

Cookies

1015. A "cookie" is a mechanism used by web browsers to make up for the "stateless" nature of the HTTP protocol. Normally, a web server has no idea that a particular user has interacted with it in the past. Cookies enable some level of 'history' to be maintained between the browser and the web server by allowing small files to be 'attached' to the browser by visited sites. On return, the site or network that uploaded the cookie can retrieve or modify the information. Cookies cannot be used to gain access to the local file system; they can only store information that the user has provided to a remote site at some point.

1016. Cookies are generally used to store information such as the passwords used to access a particular site (eg a web-based mail service), the items that have been added to a 'shopping cart' on some commercial sites, or a history of search terms that have been submitted to internet search engines.

1017. Cookies have a simple access control system that generally guarantees that only the site that installed the cookie on a browser can retrieve or modify it. However, the access controls can be set by the source to be slightly more open, allowing many more web sites to query a particular cookie. For example, a site called 'news.myportal.com.au' may submit a cookie to a browser, and set the access controls to 'Any myportal.com.au site'. Thus, other sites in the myportal.com.au domain, such as 'shopping.myportal.com.au', can query the cookies. Domain name subversion may also allow an attacker to pretend to be a site that is normally allowed to send and retrieve cookies, and so steal information.

1018. Be aware that if the same password is used for a remote site as is normally used on an internal computer system, and cookies are enabled, then the remote system can retrieve the password from the cookie and possibly use the information to penetrate the internal computing system. Many browsers offer the user the option to request confirmation before retrieving or sending cookies. Most will also have the option to disable cookies completely. Some sites rely on the cookie capability however, so disabling the capability may restrict the legitimate activity of the organisation’s users.

Applications and Plug-Ins

1019. Users who retrieve information from the Internet should be aware that the user-friendly features of the client browser can often be a danger to the user's information, or to the system on which the browser is running. In particular, the configurable application capability allows an application to be automatically launched upon download of a particular file format. An example of this would be to set up the Adobe Acrobat reader on receipt of a "PDF" file. If applications that allow a scripting language, command shell, or macro capability are configured to automatically

execute or load a downloaded file or application, the user or computer is vulnerable to attack. Often the user will be totally unaware that an attack has occurred. Some examples of applications that offer capabilities that may potentially be detrimental in an automatic-download situation are:

a. csh/sh for Unix, or command.com for DOS/Windows.

b. The Microsoft Office series of products (with macro capability or VBScript enabled).

c. Wordperfect 9.0 (with VBScript enabled).

d. Perl, TCL, Postscript interpreters or Rexx.

1020. Depending on the site risk evaluation, security administrators may decide to analyse the implications of including specific applications in the browser's automatic viewer configuration.

Java

1021. Java comes in three general forms, Java applications, Java applets, and JavaScript. Despite the similarity of the names, Java and JavaScript are actually very different. JavaScript is a series of extensions to the HTML language. It has very limited capabilities which include manipulating elements of a form, changing colours of elements of an HTML page, or opening or closing windows. The security risks associated with JavaScript are generally limited to denial of service attacks such as excessive load on the processor, or annoyance attacks. However, older implementations of JavaScript-capable browsers have included bugs that can infringe a user's privacy, and the confidentiality of their data. Examples are:

a. Reading arbitrary files from the user's computer.

b. Examining files in a browser's cache.

c. Monitoring the user's session.

d. Sending email without the user's knowledge.

e. Examining the information within a user's browser preferences, such as email address or default passwords for ftp.

1022. In contrast, the Java language is a fully featured interpreted programming language that can perform virtually any function on a computer, from playing sounds to formatting a hard drive. The security risks associated with Java applications (as opposed to Java applets) are exactly the same as normal executable code that is downloaded over the network.

1023. A Java applet, however, can be considered a Java application that has some security-related limitations imposed on it. Java applets are downloaded to the local computer, and run within a Java 'sandbox' created by a web browser. The sandbox is a security manager imposed by the web browser, which limits the Java applet's ability to perform some functions which could be considered risky. A Java applet

may not:

a. Execute arbitrary system commands.

b. Write to the local file system outside of strictly designated areas (this is browser dependent).

c. Open system libraries.

d. Open network connections to sites other than the one from which the applet originated.

1024. The functionality offered by the Java applet security manager negates many of the risks associated with downloading and running executable code. The Java applet interpreter is significantly better from a security point of view than any of the other common active content systems being used, which implement little or no security mechanisms. However, bugs in the implementation of the applet security manager, or in the function known as the bytecode interpreter, have been discovered in various platforms. These bugs can impact upon the integrity, privacy and confidentiality of information systems. For instance, bugs have been discovered that may shut down the client computer, execute arbitrary machine instructions on the client computer, create a network connection to arbitrary hosts, or otherwise bypass the applet security manager.

1025. Java and JavaScript can be disabled on most browser implementations, or the browser can be set to accept only those applets that have been digitally signed by a trusted authority. Be aware that restricting applets and JavaScript may significantly impair the functionality offered to web clients, a factor to consider when performing a risk assessment.

ActiveX

1026. Like Java applets, ActiveX controls may be included within a web page. The control is downloaded and executed on the browser's computer in the form of a pre-compiled executable. Unlike Java, ActiveX does not enforce any form of security management technology, and the ActiveX control therefore has the same level of control over the client computer as the user that is executing the browser. All risks associated with running native executables on a computer also apply to ActiveX.

1027. Whereas digital certification is an option in Java applets, some browsers enforce a system of digital signatures on ActiveX controls to ensure the user is aware of the author, and the signing authority that issued the certificate in question. If the user trusts the process used by the signing authority to certify ActiveX controls, then there can be a reasonable level of trust that the control in question performs as the author claims. However, there have been examples where 'trusted' authorities on the Internet have certified a potentially dangerous ActiveX control.

1028. Security of ActiveX controls is totally in the hands of the user running the web browser. As such, unless the user has appropriate training and is aware of the security risks associate with ActiveX controls, it may be appropriate to consider disabling this functionality within web browsers. Again, active content is a legitimate part of many web sites, and removal of this feature may impair on the functionality

that some members of an organisation rely on to perform their normal work roles.

Downloading Executables

1029. Any executable code downloaded from the Internet, whether it be via the world wide web, news groups, gopher, ftp or email, may contain malicious instructions designed to introduce a virus into the organisation, conduct denial of service attacks, or gain access to otherwise restricted information.

1030. Without appropriate controls on the download and execution of machine executable code, an organisation may be accepting significant risk to the stability and integrity of their internal network. Controls such as virus checking software may be appropriate for some organisations. Others may wish to remove executable content using a proxy or firewall solution.

Web Server Security

1031. A web server is a conceptually simple piece of software that responds to requests for information with pages formatted in the Hypertext Markup Language (HTML). An extremely simple text-only web server can be implemented on a Unix machine in literally 4 lines of code, with an extra two lines in two separate configuration files to provide the network integration. Other web servers run into hundreds of thousands of lines of code. A more complex web server component is more likely to contain errors, and some of these errors may potentially impact on the security of the server in question. In a system similar to virus checking software, keeping up to date with the most recent security patches for a web server will usually provide protection against a majority of well known attacks. However, the primary factor in web site security is the appropriate application of server configuration and management techniques.

1032. As mentioned previously in this document, a web server offers a 'window' into an organisation - a window that can be used to impact upon the integrity, confidentiality and privacy of the network, computers and users. Some simple operating-system configuration mechanisms can be used to reduce the effectiveness of potential attacks against a server. Many of the controls are operating system specific, and can be broadly grouped into the following categories:

a. ***Privilege reduction***. Running the web server as a non-privileged user, which has limited access to system resources.

b. ***File system limitation***. Ensuring that the user that runs the web server has limited access to the host file system. On Unix machines, this may imply that no symbolic links within the web tree point outside that tree, or the server runs in a 'chroot' environment.

c. ***Limited interactive system access***. Removal of non-administrative users from the computer that runs the web server further decreases the risk of circumventing any web server level access controls, may decrease the risk of accidental publishing of sensitive information, and may reduce the requirement for strong global file system access controls or comprehensive audit.

1033. Active server content is used by many web servers to enhance the functionality available to web browsers. Active server content is generally used when the web server needs to respond interactively to user input such as search engines and sites that offer online purchasing. Active server-side content can take many forms. Some of the more popular are CGI scripts, ASP, PHP, Java Servlets or Server side includes.

1034. Unlike client side Java, JavaScript or ActiveX, active server content executes locally on the machine that runs the web server. Each script that is installed presents a potential attacker with another opportunity to discover exploitable errors, and possible compromise of the computer system. Poorly written active content, or active content that is deliberately configured to provide a security hole can:

a. Be fooled into executing commands on the local system, which provides an attacker with a wedge into the organisation.

b. Leak information about the web server system that may give an attacker enough information to break into the computer.

c. Use significant processor instructions, memory or disk space as part of the run cycle, and hence be party to a denial of service attack.

1035. Active content is one of the biggest potential dangers to any web site's security. If a risk assessment identifies that web servers are a valuable resource, active content checking should be one of the first things the site should implement. Wherever possible, active content should be examined prior to installation by a programmer who is aware of the associated risks. Sites may wish to review audit logs to scan for attempts to subvert active content. A small one-line example of a simple audit program is available from **Annex B**.

Web Server Auditing

1036. Audit logs produced by a web server can be advantageous from both a security point of view, and also for usage statistics. Enabling audit in a web server is rather like operating system audit - it is not worth configuring unless there is a tangible plan for the data that is produced. Managing audit logs is a non-trivial task that usually requires ongoing maintenance and monitoring, as discussed in **Handbook 13 – Intrusion Detection & Audit Analysis**. Some traps to be wary of include:

a. File systems need to be watched to ensure that disk space does not fill, and therefore contribute to a denial of service situation.

b. Audit information needs to be protected by access controls so that it cannot be easily overwritten.

c. Audit information needs to be archived to offline storage, or removed from the file system when no longer required.

d. Audit analysis software needs to be written, installed, and run according to an agreed schedule.

e. The results of audit analysis need to be distributed to those who have a need to know, and can recognise anomalous events.

1037. The auditing of web servers can result in significant benefits to the organisation, such as:

a. Logs can identify the source of some hacking attempts, or denial of service attacks.

b. Logs can pinpoint problems with the web server configuration.

c. Usage statistics can identify when an upgrade of network bandwidth is likely to be required.

d. Usage statistics can identify the most or least popular pages on a site.

e. If access controls are in use on a web server, audit logs can be distributed to owners of the data within the restricted area, in order to distribute the audit analysis capability, and potentially identify situations where unauthorised users have access to sensitive data.

1038. A site that wishes to guard against denial of service attacks that attempt a large number of connections to the web server with a goal of filling the file system of the local machine may wish to consider locating the audit logging facility on a separate physical or logical disk device. If this is not possible, a form of automatic log rotation may be appropriate. In the same way that web server auditing allows the examination of users on a web server, web proxy auditing or firewall auditing will allow a site to examine the activities of internal users when browsing external networks such as the Internet. Again, the analysis and management of audit logs is not a trivial task.

Protecting Sensitive Information

1039. Authentication, access controls and encryption facilities found in commercial web browser and server software can be used to provide need-to-know access controls within an organisational network. However, whenever Government classified information is being transmitted over a network of lower classification, these facilities may not be appropriate, and sites should use DSD approved encryption as per the recommendations found in **Handbook 9 - Cryptographic Systems**.

1040. Access control facilities fall into four broad categories:

a. IP address or Domain Name access control.

b. Proxy server access controls.

c. Password Authentication.

d. Encrypted Transactions and Public Key Infrastructure.

1041. Access control by IP address or domain name is generally not secure against a determined attacker, and thus is not recommended unless combined with

some other authentication mechanism. IP addresses or domain names can be forged quite simply. A proxy server is a URL filtering device which is generally employed as a local cache to speed effective external web access to organisational browsers. In combination with an effective firewall, a proxy server can also be configured to limit browsers on an external network to access only a subset of an organisation's internal web pages using access control lists. In combination with public key infrastructure and client and server certificates as mentioned below, a proxy server can also allow pass-through access to such restricted pages to cryptographically authenticated external browsers. Password authentication for web servers is susceptible to the same problems that plague normal operating system passwords such as network interception and replay, exhaustive password attempts, and dictionary attack if the attacker has access to the web server configuration files.

1042. However, web servers do not have the same security functionality associated with passwords as is normally found within an operating system, such as disabling an account after a series of incorrect password attempts, or audit log integration for failed passwords. In addition, a web browser will often cache access passwords, and automatically re-authenticate when a server requests a password within a set timeout period. Password authentication may be an appropriate solution for access control needs if the browser and server are both operating on a trusted network, and the passwords do not pass over a network which is not under the control of the organisation. Passwords may be stored within the web server's configuration files, or within a separate directory server - which may or may not be integrated with operating system user name and passwords.

Encrypted Transactions and Public Key Infrastructure

1043. Several proposed standards exist for web-based encryption. Only one, Secure Sockets Layer (SSL), has gained significant market penetration. The SSL protocol implements several security provisions including server authentication, client authentication, and data encryption. SSL relies on an asymmetric encryption scheme based on X.509 certificates to negotiate a symmetric secret key that is then used to encrypt data between client and server. SSL normally relies on a web of trust created by a Public Key Infrastructure (PKI) in order to work effectively. The use of cryptography in Government system, and specifically the need to use evaluated cryptographic products, is detailed in **Handbook 9 - Cryptographic Systems**.

1044. Sites that wish to use a web-based encryption system to provide need-to-know controls should use keys / certificates from approved Government Public Key Infrastructure (GPKI) facilities. This is further discussed in **Handbook 11 - Email Security.** Sites may set their web server software to apply access controls on either the data contained within the certificate, such as department or user name, or the metadata within a X.500 or LDAP directory server that is associated with a unique identifier such as a user ID that is supplied on the certificate. Examples of such metadata may be 'groups' that align closely with departmental membership or area of responsibility.

1045. Client certificates may be stored in a number of different configurations such as:

a. Within a user's home directory, relying on operating system access controls to determine access to the digital certificate, and providing a single-sign-on facility to a site's user base for access to the normal operating system desktop and the protected web.

b. Within a user's home directory, but protected using password-based authentication over and above the normal operating system access controls.

c. Within a directory server, and optionally protected using additional password authentication.

d. On magnetic media for ease of transport, and optionally protected by additional password-based authentication.

e. On a smart card, and optionally protected by additional password or biometric authentication.

1046. Ignoring physical security issues, the items above are ranked approximately in order of security. Less auditing and access control management is required for each level in order to attain a similar level of assurance. For example, significant auditing and access control would be required to bring home-directory based certificate storage to a similar security level as a smart card protected by biometrics.

Grades of Web Server and Client Security

1047. The following grades of Web Server and Client Security implementation have been included to assist in determining the level of effort that should be allocated to such a task. They are not definitive, and when implementing security should be used as a guide only. Agencies should note that this information does not replace the requirements of the **Gateway Certification Guide**. Implementation of web security will vary from organisation to organisation. A risk assessment should be performed for the organisation, with emphasis on web issues, before implementing any of the recommendations displayed below.

a. **Grade 0**

Web Servers

i. Active server content added to web servers based on risk assessment.

ii. Regular application audit to ensure web client and server software is upgraded with the latest security patches.

iii. Web servers are set to run as a user with minimal file system or operating system privileges.

Users and Web Clients

i. Users to be informed of the dangers associated with web security.

b. **Grade 1**

Web Servers

i. Active server content added to web servers based on risk assessment.

ii. Regular application audit to ensure web client and server software is upgraded with the latest security patches.

iii. Web servers are set to run as a user with minimal file system or operating system privileges.

iv. File-level access controls on the web server operating system restrict directory trees that contain controlled information to authorised staff only.

v. Formal checking process used to verify any custom or freeware active server content that is to be added to web servers. Commercial active content to be added based on risk assessment.

vi. Audit is configured on all web servers, and rotated nightly to assist with problem diagnosis and repair.

Users and Web Clients

i. Users to be informed of the dangers associated with web security.

c. **Grade 2**

Web Servers

i. Active server content added to web servers based on risk assessment.

ii. Regular application audit to ensure web client and server software is upgraded with the latest security patches.

iii. Web servers are set to run as a user with minimal file system or operating system privileges.

iv. Accounts on web servers that implement access controls are restricted to administrative users only, or file-level access controls on the operating system restrict access to directory trees that contain controlled information.

v. Formal checking process used to verify any custom or freeware active server content that is to be added to web servers. Commercial active content to be added based on risk assessment.

vi. Audit is configured on all web servers, and analysed for general intrusion attempts, or attempts to circumvent access control lists.

Audit logs are either rotated, or migrated to offline storage depending on results of risk assessment.

vii. Proxy server configured to virus check all incoming executable code.

viii. Formal sanitisation and checking process used to transfer information from the internal network to the internet web server in order to guard against sensitive information leakage.

ix. Firewall installed to direct all outgoing web requests to an organisational proxy server.

x. In situations where access controls are implemented, either:

Public key based encryption based on x.509 certificates is used between server and client to enhance data confidentiality.

Password-based access controls are used, and staff are informed of the requirement to choose appropriate passwords that are not easy to guess.

IP Address access controls are used, and appropriate network change control and intrusion detection mechanisms are in place to minimise the risk of IP address spoofing.

Users and Web Clients

i. Users to be informed of the dangers associated with web security.

ii. Users' browsers configured to reject all Java and ActiveX.

c. **Grade 3**

Web Servers

i. Active server content added to web servers based on risk assessment.

ii. Regular application audit to ensure web client and server software is upgraded with the latest security patches.

iii. Web servers are set to run as a user with minimal file system or operating system privileges.

iv. Accounts on the web servers are restricted to administrative users only.

v. Formal checking process used to verify any custom or freeware active server content that is to be added to web servers. Commercial active content to be added based on risk assessment.

vi. Audit is configured on all web servers, and analysed for general intrusion attempts, or attempts to circumvent access control lists. Audit logs are either rotated, or migrated to offline storage depending on results of risk assessment.

vii. Proxy server configured to virus check all incoming executable code.

viii. Formal sanitisation and checking process used to transfer information from the internal network to the internet web server in order to guard against sensitive information leakage.

ix. Firewall installed to direct all outgoing web requests to an organisational proxy server.

x. In situations where access controls are implemented, either:

Public key based encryption based on x.509 certificates is used between server and client to enhance data confidentiality.

Password-based access controls are used, and staff are informed of the requirement to choose appropriate passwords that are not easy to guess.

Users and Web Clients

i. Users to be informed of the dangers associated with web security.

ii. Users' browsers configured to reject all Java and ActiveX.

iii. Users' browsers configured to reject all JavaScript and Cookies if determined appropriate by risk assessment.

iv. Application extensions to user browsers are evaluated by qualified system security staff if determined appropriate by risk assessment.

v. Proxy server or firewall installed and configured to strip all non-essential information from outgoing web page requests such as referral information or browser version.

c. **Grade 4**

Web Servers

i. Active server content added to web servers based on risk assessment.

ii. Regular application audit to ensure web client and server software is upgraded with the latest security patches.

iii. Web servers forced by the operating system to use a virtual root - a subset of the computers file system from which the web server cannot escape.

iv. Accounts on the web servers are restricted to administrative users only.

v. Formal checking process used to verify any custom or freeware active server content that is to be added to web servers. Commercial active content to be added based on risk assessment.

vi. Audit is configured on all web servers, and analysed for general intrusion attempts, or attempts to circumvent access control lists. Audit logs are either rotated, or migrated to offline storage depending on results of risk assessment.

vii. Proxy server configured to virus check all incoming executable code.

viii. Proxy server or firewall installed and configured to strip all incoming information outside the scope of the official HTML specification, such as executables or JavaScript

ix. Formal sanitisation and checking process used to transfer information from the internal network to the internet web server in order to guard against sensitive information leakage.

x. Firewall installed to direct all outgoing web requests to an organisational proxy server.

xi. In situations where access controls are implemented, public key based encryption based on x.509 certificates is used between server and client to enhance data confidentiality.

Users and Web Clients

i. Users to be informed of the dangers associated with web security.

ii. Users' browsers configured to reject all Java and ActiveX, JavaScript and Cookies.

iv. All application extensions to user browsers are evaluated by

qualified system security staff.

v. Proxy server or firewall installed and configured to strip all non-essential information from outgoing web page requests such as referral information or browser version.

ANNEX A

Privacy Issues in Browser Operation

Three companies, ABC Inc, RST Pty Ltd and XYZ Co. have all submitted tenders to the Department of TVW to supply the same products - Flat screen computer monitors.

One of the members of the team at the Department who is examining the contracts wishes to find out some more details about the product range, and so logs in to TVWWeb, and performs a search on "(ABC OR RST OR XYZ) AND "flat screen" on one of the major search engines.

The search engine site will now have a log of the fact that "browser.tvw.gov.au" (ie the theoretical TVWWEB machine used by the tender team) has performed a search on the items "ABC, RST, XYZ and flat screen".

The user clicks on the search engine result that points to the ABC Inc web page describing their Flat screen computer monitors.

ABC Inc. will now have a log entry that looks something like this:

```
browser.tvw.gov.au - -
[13/Aug/1999:14:25:05 +1000] "GET
/products/flatscreens.html
HTTP/1.0" 200 1209 "
http://www.asearchengine.com.au/query?(
ABC|RST|XYZ)&flat.screen" "Mozilla
4.6 [en] (WinNT;I)"
```

The log entry is broken down roughly as follows:

Log Entry	Explanation
Browser.tvw.gov.au	The name of the host which made the request
13/Aug/1999:14:25:05	The date and time the request was made
GET /products/flatscreens.html	The page that the user requested
1209	The number of bytes transferred
http://www.asearchengine.com.au/query?(ABC|RST|XYZ)&flat.screen	The previous link that was visited by this user. In this case, the information informs the site that the user visited a search engine, and asked for information on flat screen technology from ABC, RST or XYZ.
Mozilla 4.6	The user is using version 4.61 of the Netscape browser
[en]	The user has the English language version of the browser
WinNT	The browser is running on a Windows NT platform

The user hits the 'back button' on their browser, scans the entries for the flatscreen page for RST Pty Ltd, and selects the link.

RST Pty Ltd will now have a log entry similar to the one received by ABC.

The user hits the 'back button' on their browser, scans the entries for the flatscreen page for XYZ Co., and selects the link.

XYZ Co. will now have a log entry similar to the one received by both ABC and XYZ

Each of the companies now has audit records which indicate with a reasonable level of confidence that there are at least two competitors who may be applying for the flat screen contract at the Department of TVW. The logs even detail the competitors' names. This may give each company enough information to amend their tender application to better compete against the other applicants.

As the staff that handle the tender conclude their deliberations, and the day of the announcement draws near, ABC Inc. is chosen as the sole supplier. The tender team distribute the information via the internal web (intranet) to a small group of managers who have a need to know for the information. On the internal limited distribution web page that describes the chosen supplier (tenderwinner.html), there is a link to the flat screen product information from the ABC web page. Each of these managers selects the link to find out more information.

ABC Inc now has a large surge of activity on their web site - all originating from TVWWeb. The logs look something like this:

```
browser.tvw.gov.au - -
[16/Aug/1999:14:25:05 +1000] "GET
/products/flatscreens.html
HTTP/1.0" 200 1209
"http://www.internal.tvw.gov.au/tenders/te
nderwinner.html" "Mozilla/4.6 [en]
(WinNT;I)"
manager1.tvw.gov.au - -
[16/Aug/1999:14:29:05 +1000] "GET
/products/flatscreens.html
HTTP/1.0" 200 1209
"http://www.internal.tvw.gov.au/tenders/te
nderwinner.html" "Mozilla/4.6 [en]
(WinNT;I)"
manager2.tvw.gov.au - -
[16/Aug/1999:14:30:05 +1000] "GET
/products/flatscreens.html
HTTP/1.0" 200 1209
"http://www.internal.tvw.gov.au/tenders/te
nderwinner.html" "Mozilla/4.6 [en]
(WinNT;I)"
manager3.tvw.gov.au - -
[16/Aug/1999:14:32:05 +1000] "GET
/products/flatscreens.html
HTTP/1.0" 200 1209
"http://www.internal.tvw.gov.au/tenders/te
nderwinner.html" "Mozilla/4.6 [en]
```

(WinNT;I)" Unfortunately, ABC Inc. now has a log on their server, which points towards the possibility that they have succeeded in winning the tender - prior to the official notification date. The fact that there is significantly increased activity from 'tvw.gov.au' just prior to the notification date is a good indication. The fact that the referral document had a file name of 'tenderwinner.html' increases the possibility even more.	

ANNEX B

Simple Access Log Analysis for Unix

The code below illustrates a mechanism to ensure that users are not attempting to subvert a particular custom CGI-BIN script on a web server.

The 'mycgi.pl' program takes an argument composed of valid alphanumeric characters only. Any attempts to send the script non-standard characters may be an attempt by a user to subvert the security of a system.

```
cd /var/log/httpd; cat access_log | cut -d\" -f2 | egrep -v "^(get|GET) /cgi-bin/mycgi.pl\?[a-zA-Z0-9]* (http|HTTP)/[0-9]\.[0-9]$"
```

The code in question runs through the web server access log, cuts out all but the URL request, and then compares the request against the legitimate request format. Any attempts to access the CGI program in a non-standard way will display on the screen.

Such audit aims can be incorporated in more comprehensive audit analysis code.

ANNEX C

Web Server Security Analysis

As stated previously, web server vulnerabilities can exist in a number of areas. This means that an examination of web server security needs to cover a broad range of areas. The following questions should be asked when assessing the security:

1. What functions are performed by the web server?

2. What software is installed?

3. Does the web server contain on on-line database or access a remote database server?

4. What data is collected by the web server?

5. Does the web server make use of active content?

6. What scripts are run on the web server?

7. Who is responsible for the following activities:

- content management,
- network security,
- operating system security and maintenance,
- application level review, maintenance and security,
- auditing and monitoring?

8. Who are the data/system owners?

9. What disclaimers/privacy statements are published?