

StatSoft®

Business White Paper

WebSTATISTICA Security

Last Updated: June 2003

U.S. Headquarters: StatSoft, Inc. • 2300 E. 14th St. • Tulsa, OK 74104 • USA • (918) 749-1119 • Fax: (918) 749-2217 • info@statsoft.com • www.statsoft.com

Australia: StatSoft Pacific Pty Ltd.
Brazil: StatSoft Brazil Ltda.
Czech Republic: StatSoft Czech Rep. s.r.o.
France: StatSoft France

Germany: StatSoft GmbH
Hungary: StatSoft Hungary Ltd.
Israel: StatSoft Israel Ltd.
Italy: StatSoft Italia srl

Japan: StatSoft Japan Inc.
Korea: StatSoft Korea
Netherlands: StatSoft Benelux BV
Poland: StatSoft Polska Sp. z o. o.

Portugal: StatSoft Iberica Ltda.
Russia: StatSoft Russia
Singapore: StatSoft Singapore
S. Africa: StatSoft S. Africa (Pty) Ltd.

Spain: StatSoft Espana
Sweden: StatSoft Scandinavia AB
Taiwan: StatSoft Taiwan
UK: StatSoft Ltd.

Table of Contents

Executive Summary	3
Introduction to <i>WebSTATISTICA</i>	3
System Architecture.....	3
User Authentication.....	5
Identity Tokens.....	6
User Account Management	6
Web Based User Management	7
Permissions While in the Software.....	7
Consumers of Information.....	7
Users Who Perform Analyses	8
Users Who Can Deploy Scripts.....	8
Administrators.....	8
Access Control.....	9
Special Considerations for Network Shares	9
Communication Between Computers	10
Web Server/Client Communication.....	10
Web Server/ <i>STATISTICA</i> Server Communication.....	10
PHP Security.....	11
IIS Security.....	11
Updates.....	11
Sample Applications	12
Unused ISAPI Extensions	12

Executive Summary

WebSTATISTICA Server provides Web-based access to **STATISTICA** analyses. In today's Internet world, the security of any Internet application is of primary consideration. This white paper addresses this concern as it relates to **WebSTATISTICA Server**. After an introduction and system overview, the security topics that will be discussed include user authentication, user management, access control, user permissions, and communication between computers. Included is information involving security issues with PHP, the widely used scripting language on which the Web server component of **WebSTATISTICA Server** depends. We will also discuss security issues involved when the Web server used is Microsoft Internet Information Services (IIS).

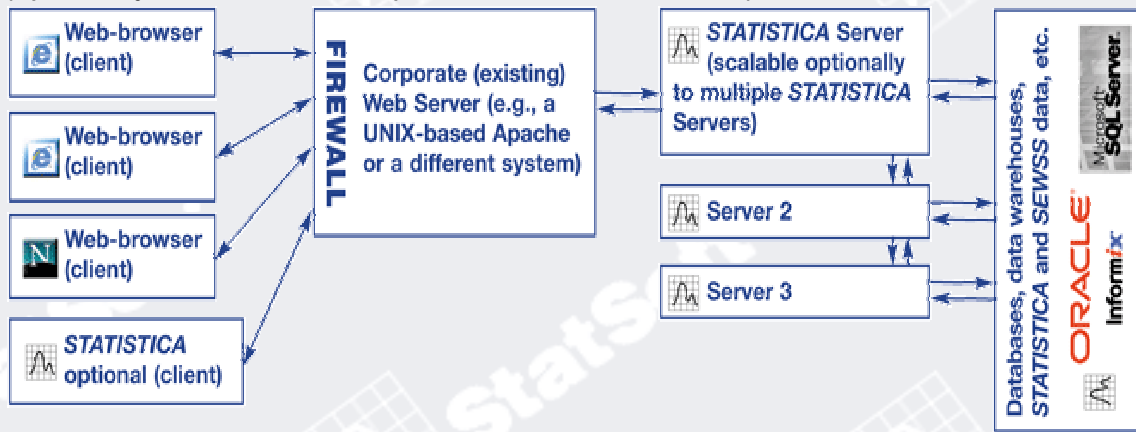
Introduction to **WebSTATISTICA**

WebSTATISTICA Server is a highly scalable, enterprise-level, Web-based data analysis and database gateway application system that is built on distributed processing technology and fully supports multi-tier Client-Server architecture configurations. **WebSTATISTICA Server** exposes the analytic, query, reporting, and graphics functionality of **STATISTICA** through easy to use, interactive, standard Web interfaces. It is offered as a complete, ready to install application with an interactive, Internet browser-based ("point-and-click") user interface, enabling users in remote locations to interactively create data sets, run analyses, and review output. However, because of its open architecture, **WebSTATISTICA Server** also includes development kit tools (based entirely on industry standard syntax conventions such as VB Script, C++, HTML, XML, JavaScript, and Java), that enable IT department personnel to customize all main components of the system or expand it by building on its foundations, for example, by adding new components and/or corporation-specific analytic or database facilities. The system is compatible with all major Web server software platforms (e.g., UNIX Apache, Microsoft IIS), works in both Microsoft .net and Sun/Java environments, and does not require any changes to the existing firewall and Internet/intranet security systems.

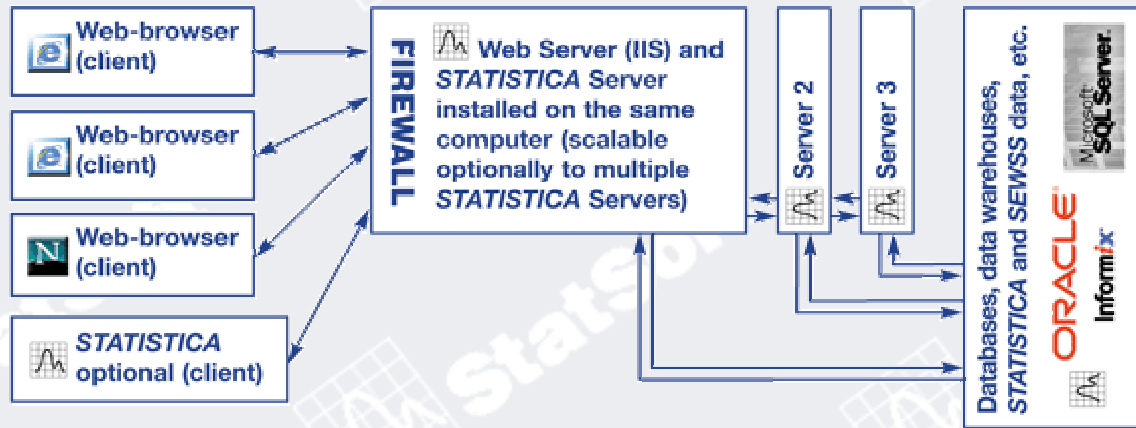
System Architecture

The **WebSTATISTICA** architecture uses two different servers: the industry-standard Web server (IIS, Apache, etc.), which handles the traffic from the browser, and the **STATISTICA Server**, which processes all the analysis requests.

Although the general design uses two "machines" in a typical configuration, the Web server (e.g., a UNIX-based Apache system) and at least one **STATISTICA** Server (optionally scalable to multiple **STATISTICA** Servers),



in many cases, the **STATISTICA** Server is installed on the same machine (for example, when IIS is used as the Web host).



The design allows for a flexible, generic Web server implementation by using a standard scripting language on the Web server. The purpose of the Web server is to package requests from the user (received from a browser), send these to the **STATISTICA** Server, and then process responses from the **STATISTICA** Server for display to the user (on his/her browser).

Communication between the Web server and the **STATISTICA** Server is accomplished through technology based on the industry standard XML conventions. The system is fully customizable, and customers who want to develop their own modifications or extensions of this (ready to deploy) system can use the development tool kit facilities provided to modify all aspects of the scripts that are being executed by **STATISTICA** (on the **STATISTICA** Server side) and the appearance of the user interface exposed to the end-users on the (browser-based) thin client side. Only the

most standard, commonly known tools (such as VB or XML/HTML) are used to customize or expand the system.

The actual Web page definitions and **STATISTICA** scripts to be executed are stored in a designated Repository Facility on the **STATISTICA** Server, and they are managed in a queue-like fashion. The system also includes a highly optimized Distributed Processing Manager that handles the incoming processing load and distributes it optimally over multiple threads of **STATISTICA** and multiple **STATISTICA** Server computers.

The **WebSTATISTICA Server** software system also includes the **STATISTICA** Visual Basic Web Extensions. These extensions to the SVB language enable the script writer either to let the system take care of displaying the resulting graphs and spreadsheets on the automatically generated (output) Web pages, or to customize the appearance of the generated output pages by adding HTML directives as appropriate.

Security and authentication is a key design feature in the **WebSTATISTICA Server** application system. At the beginning of the session, users "sign on" to the system with their user names and passwords. System administrators are able to control access to data sources and scripts based either on user or group permissions. The highest level of the access privilege allows advanced users (or administrators) to execute virtually arbitrary scripts (e.g., in order to perform system administration or maintenance operations). That level requires a designated (highest) access privilege because, due to the general nature and power of the **STATISTICA** Visual Basic language, it gives access (to the authorized users) to all resources on the network.

Note that this system can be integrated with the "traditional" (i.e., non-Web-based) **STATISTICA** concurrent network or a **STATISTICA** enterprise system authentication scheme so that a corporate customer can install, for example, a 50-user (total) **STATISTICA** enterprise system or a concurrent network with 20 licenses accessible via the **WebSTATISTICA Server**.

User Authentication

WebSTATISTICA Server uses the NT security model for both user authentication and for restricting access to files in the **WebSTATISTICA** Repository. Access to the server is contingent on membership in local groups that are created by **WebSTATISTICA** during installation. Since **WebSTATISTICA** uses the NT security model, it can be integrated with existing Domain and Active Directory network configurations.

There are two modes of authentication supported by **WebSTATISTICA**. By default, a user is always required to enter an account name and password to log on to the server. The password is encrypted by the client before it is submitted to the server. The encryption algorithm is designed to keep casual examination of network traffic by network sniffing devices from easily displaying plaintext passwords. While this protection may be sufficient in many circumstances, an increased level of security can be achieved by using the Secure Sockets Layer (SSL) protocol, which is described later in this document. Installations that use Internet Information Services (IIS) on a local intranet to host the **WebSTATISTICA** Web files can additionally enable Integrated Login. When Integrated Login is configured, an attempt is made to authenticate you by passing your network security context to the **WebSTATISTICA** Server. If the detected account is a member of the appropriate local groups, you will be passed through without the need to enter the information manually. No password information is submitted by the client when Integrated Login is used.

Identity Tokens

Once a user is authenticated, a unique identifier called an Identity Token is created. The Identity Token is what represents the session; any further communication between the client and the server will include this identifier. Once a session is ended, the corresponding Identity Token is no longer valid and cannot be reused. A session ends when the user selects to be logged off the server or when the session times out. A session will time out if a user closes the browser window or browses to a new page and does not return to the **WebSTATISTICA** interface within the time-out period. The amount of time it takes a session to expire is configurable by **WebSTATISTICA** administrators.

User Account Management

Access to the **WebSTATISTICA Server** is determined by local groups that exist on the machine on which the **STATISTICA Server** is installed. The minimum requirement to access any resources on a **WebSTATISTICA Server** is to be a member of the local group SWS_USER; every **WebSTATISTICA** account is a member of this group. Membership in the **STATISTICA Server's** local group Users is not required to log on to **WebSTATISTICA**. Local and domain accounts can be members of SWS_USER. When local accounts are used, the account password is validated against the **STATISTICA Server's** local account database. Domain accounts are validated against the appropriate domain controller. To ease administration in a domain environment, it is common for a global group to be added to the local group SWS_USER. This enables user management to be performed using traditional domain management tools instead of needing to add users directly to the local group

on the **WebSTATISTICA Server**. The same thing can be done with the other **WebSTATISTICA** local groups, for example, the SWS_ADMIN group may have only one member, SWS_ADMIN_GLOBAL, which is a global group contained within the Active Directory.

Web Based User Management

While traditional Windows tools can be used to manage **WebSTATISTICA Server** accounts, you are also provided with a Web-based interface for managing these same accounts. **WebSTATISTICA** administrators can make use of this Web-based interface by logging on to the **WebSTATISTICA Server**. From this interface, they can create new users or add existing NT accounts to the **WebSTATISTICA Server** user pool. Additionally, administrators can create **WebSTATISTICA** user-defined groups, which are simply local groups on the **STATISTICA Server** that begin with the prefix SWSU_. These groups are convenient for grouping together related users when defining who may access specific documents in the Repository.

Permissions While in the Software

Every **WebSTATISTICA** user must be a member of the local group SWS_USER. There are several other local groups created by the installer in addition to SWS_USER that can be used to assign roles to **WebSTATISTICA** users. Different roles a user might have include consumers of information, users who perform analyses, users who can deploy scripts, and administrators.

Consumers of Information

Often there are users who are not interested in performing the analyses themselves but instead are interested in viewing already prepared reports and drilling down on the information further. The product appropriate for this type of user is **WebSTATISTICA Knowledge Portal**. **WebSTATISTICA Knowledge Portal** enables your colleagues, employees, and/or customers (with appropriate permissions) to log on and quickly and efficiently get access to the information they need by reviewing predefined reports (optionally organized into structured repositories). Additionally, the **WebSTATISTICA Interactive Knowledge Portal** offers options to the portal visitors to define and request new reports, run queries and custom analyses, drill down and up, and slice/dice data (optionally via OLAP), and gain insight from any resources made available to them by the portal designers or administrators.

Once an account is a member of the local group SWS_USER, they have access to all of the advanced statistical modules licensed on the **STATISTICA Server**. Two

additional groups, SWS_PORTAL and SWS_PORTALINTERACTIVE, define an account as just a portal user. Once an account is made a member of one of the portal groups, each log on is automatically redirected to the streamlined **Knowledge Portal** interface where the user is able to access the reports to which they have been given access.

Users Who Perform Analyses

There are different degrees of power that can be given to users who perform analyses. Membership in the local group SWS_BATCHABLE allows users to queue jobs to be run by the server in batch mode. Jobs run in batch mode continue to execute even when the user is not logged on to the server. A user with batch rights could potentially place a larger strain on server resources by submitting an unreasonable amount of batch jobs to be run by the server.

The local group SWS_DOWNLOADFILES defines which users are allowed to copy documents from the Repository to their local machine. The local group SWS_UPLOADDATA allows users to utilize the *Upload Document to Server* facility to place data files in the Repository from their local machines.

Users Who Can Deploy Scripts

One important restriction on the group SWS_UPLOADDATA is that it does not give the user the ability to upload SVB macros. Only users who are a member of the local group SWS_UPLOADSCRIPTS can upload this type of file. It is important that this distinction between SVB macros and other uploads be made because SVB macros run in the same context as the **STATISTICA** Server and, therefore, have full access to the same resources that the LocalSystem account does. If you give users the ability to upload scripts, you are potentially giving them full access to the server on which **WebSTATISTICA** is installed.

Administrators

SWS_ADMIN is the local group that gives an account administrative access to **WebSTATISTICA**. Administrators have the ability to control access to all items in the Repository. Since all other group membership is implied when a user is a member of SWS_ADMIN, an administrator is able to create batch jobs, upload/download data files and upload scripts even if they aren't explicit members of those groups. If **Knowledge Portal** or **Knowledge Portal Interactive** users are made members of SWS_ADMIN, they will no longer be considered portal users since that is considered a restriction in access.

Access Control

The Repository is the main data store used in **WebSTATISTICA**. All user data, configuration files, results pages, analysis and user interface script files, and other kinds of information reside there. The Repository sits on top of the NTFS file system. Because **WebSTATISTICA** uses an integrated security model, it uses NTFS to enforce access to files stored within the repository. Restricting users from accessing specific files is as easy as changing the NTFS permissions. Additionally, the **WebSTATISTICA** Repository Manager goes beyond the NTFS permissions and ensures that users cannot modify system files they are not supposed to, even if the NTFS permission allows it.

A powerful Web-based interface is provided to **WebSTATISTICA** users that enables them to define NTFS permissions for files and directories to which they have access. For regular users, this would be their user directory and items they have access to in the Shared folder. Administrators can define permissions to other users' documents as well. It is recommended to use the Web-based interface to control these NTFS permissions because some of the complexity of required permissions is managed automatically. If permissions are modified outside of the **WebSTATISTICA** interface, keep in mind that LocalSystem and SWS_ADMIN should always be given full access to all items.

Special Considerations for Network Shares

It is sometimes desirable to define a network file share that points directly to the Repository so that files can be copied to a mapped drive instead of having to be uploaded through the client's Web browser. Uploading files through the Web browser is a comparatively inefficient method of transferring files and is usually only desirable when the client does not have local network access to the server. By creating a network share inside the Repository, users can take advantage of the existing network file sharing services and seamlessly share data files between the concurrent network version of **STATISTICA** and **WebSTATISTICA**. However, by allowing users direct access to the Repository, they are now able to bypass the built-in security provided by the **WebSTATISTICA** Repository Manager, including its restrictions on uploading scripts.

The solution to this security concern is to make use of the Repository directory Files\Shared. Because of special restrictions defined in RepositoryConfig.xml, it is safe to share this directory on your network. SVB macros can never be run from that directory, and if an SVB macro is placed in the directory, it will not be exposed to any user that is not a member of the local group SWS_UPLOADSCRIPTS, nor can it be copied to any other directory. No other directories in the Repository outside of the

Files\Shared directory should be made available as this would allow anyone with access to that share to run potentially malicious macros on the server.

Communication Between Computers

There are two channels of communication that can be potential security holes for a **WebSTATISTICA** installation. The first is the communication between the client's browser and the Web server, the other is the communication between the Web server and the **STATISTICA** Server.

Web Server/Client Communication

As mentioned earlier, if Integrated Login is not used, the password that the user logs on with is submitted in encrypted form to the server. This, however, is the only information that is encrypted by **WebSTATISTICA**. Sensitive results or other data could be intercepted by network packet sniffing devices. If your Web server is only available to clients inside your firewall, this may not be a concern. If your **WebSTATISTICA Server** is accessible to the public and secure communication is a requirement, the solution to this problem is to enable the Secure Sockets Layer (SSL) protocol on your Web server. When SSL is enabled on the Web server, all communication between the client and the Web server will be encrypted. To enable SSL on your Web server, an SSL certificate must be obtained from one of the certificate authorities (CAs), such as VeriSign or Thawte. Popular Web server software such as IIS and Apache support the installation of SSL certificates.

Web Server/STATISTICA Server Communication

The other potential security risk involves installations where the **WebSTATISTICA** Web files reside on a different computer than the **STATISTICA** Server. **WebSTATISTICA** uses SOAP, an XML-based protocol, to communicate between the Web server and the **WebSTATISTICA** executables. This SOAP communication is not encrypted, and as a result any communication between the client and the **WebSTATISTICA** Server will be sent in unencrypted form when it passes from the Web server to the **STATISTICA** Server. This is true even when an SSL certificate is installed on the Web server machine. This is not an issue when the **WebSTATISTICA** Web files and the **STATISTICA** Server are on the same computer, because the traffic never shows up on the network. Even if the data is sensitive, if the computers are both within your firewall, the fact that this data is not encrypted may not be an area of concern. Future plans include securing communication between the Web server and the **STATISTICA** Server when they are not on the same machine in order to accommodate cases where the Web server and **STATISTICA** Server are in two

separate physical locations. Check with StatSoft for the availability of this feature if it is a design requirement.

PHP Security

PHP is a powerful server side scripting language utilized by the **WebSTATISTICA** Web server component. It is always important to use the latest release of PHP as each new release addresses newly discovered vulnerabilities. The **WebSTATISTICA Server** installer includes the latest release of PHP as well as a PHP configuration file with the settings that are appropriate for the correct operation of **WebSTATISTICA** in a production environment.

The PHP scripts installed with **WebSTATISTICA** translate client Web browser requests into the SOAP protocol-based messages that are used to communicate with the **STATISTICA Server**. As with all aspects of **WebSTATISTICA**, we have made every effort to ensure that the scripts we distribute are as secure as possible.

The PHP online manual contains a section devoted to security considerations for PHP. Most of the information on this site relates to scripting, but there are two sections regarding installation on a UNIX/Apache server that should be given consideration.

<http://www.php.net/manual/en/security.php>

IIS Security

While the **WebSTATISTICA Server** can be hosted by virtually all Web server software, IIS is the most common choice. A new installation of IIS is not safe for a production Web server. Below are a few steps that should be taken to help ensure the safety of your server. While these recommendations will help to secure your Web server, StatSoft, Inc. makes no guarantees that following the recommendations will make your server completely secure. Responsibility for the security of your Web server is the responsibility of the customer.

Updates

There are known security vulnerabilities that can be addressed by updating your installation with the latest security patches.

For IIS 4 (NT 4) this involves applying Service Pack 6a followed by several cumulative updates.

<http://www.microsoft.com/ntserver/nts/downloads/>

For IIS 5 (Windows 2000) and 5.1 (Windows XP) the updates are included with service packs.

<http://www.microsoft.com/windows2000/downloads/>

<http://www.microsoft.com/WindowsXP/pro/downloads/>

These updates can also be installed through the Windows Update service:

<http://windowsupdate.microsoft.com>.

It is important to stay current with new security updates. Microsoft has released a tool called HFNetChk that helps system administrators with this task.

<http://www.microsoft.com/technet/security/tools/hfnetchk.asp>

Sample Applications

There are several sample applications installed with IIS that are there to demonstrate features of the software. Their location is readily known and, therefore, they are frequently targeted by hackers. These example applications are located in the directory scripts in your Web root (often `c:\inetpub\wwwroot\scripts`). These example files are not intended for a production server and should be removed.

Unused ISAPI Extensions

There are many ISAPI extensions enabled by IIS that are not needed by most Web sites. The fact that they are present makes your site vulnerable to buffer overflows that can result in denial of service attacks and even allows access to your server. Removal of these unused ISAPI extensions can be done by hand in the Internet Services Manager, or you can use the IIS Lockdown Wizard from Microsoft, a highly recommended tool. Also part of the IIS Lockdown Wizard is the URLScan filter, which can intercept maliciously formed HTTP requests that hackers use to gain access to your server.

<http://www.microsoft.com/technet/security/tools/locktool.asp>