



# Global Information Assurance Certification Paper

Copyright SANS Institute  
Author Retains Full Rights

This paper is taken from the GIAC directory of certified professionals. Reposting is not permitted without express written permission.

## Interested in learning more?

Check out the list of upcoming events offering  
"Security Essentials Bootcamp Style (Security 401)"  
at <http://www.giac.org/registration/gsec>

## **Designing a Secure Server Model**

Marshall Wells

February 7, 2001

This paper is not intended as a step by step guide to building a secure server. Most operating system vendors and many third parties have published guidelines focussed on how to secure specific operating systems. This paper is presented as an overview of the requirements for building a secure server with an emphasis on making the process reproducible. If an organization has determined to deploy servers that have been built to meet set security guidelines, it is reasonable to put in place a process for meeting that requirement. Relying on individuals to “hand-craft” servers and maintain any level of consistency is just not a reasonable expectation.

### **Why Create a Model?**

As new vulnerabilities are discovered for particular operating systems, it is necessary to evaluate their impact on deployed systems. If servers within an organization have been built to conform to a specific model, the components and their versions will be easier to ascertain. This in turn will make it easier to plan remediation of vulnerabilities that have been determined to present a risk. A formal build document, or model, can be used to accredit (recognize as conforming to a standard) a server before it is deployed. The same document or process can be used to periodically reaccredit servers. For an organization with very few servers it may not make sense to create a formal model. For organizations with many servers, a model becomes a tool for promoting supportability as well as for promoting security.

Continuity of support is an important issue in today’s information systems environment. With information systems professionals changing positions or employers at a high rate, maintaining organizational knowledge can become a challenge. Creating a model and utilizing it in the build of servers can provide a mechanism by which new employees can be more quickly brought up to speed on the technology already deployed. A well written build document can also give new employees, or employees who are fulfilling a new role, insight into the decision process involved in creating the standards for the organization.

Reproducible work is not always a given in information systems work. Sometimes it takes a tweak here and a nudge there to make a system behave as it should. If those small adjustments are not recorded, the system is not easily reproducible. If it should become necessary to quickly bring up a new web server within a short timeframe, for example, a standard web server build will be indispensable. If a model is defined, and the documentation maintained as system changes become necessary, bringing a new server into the environment should be easier. The end result will also be a system that demonstrates a higher level of compliance with organizational standards and security policies

and guidelines.

## **Know the Server Operating System**

It seems as if it is common for system administrators to look for quick fixes to securing servers. One tool commonly utilized to quickly identify vulnerabilities is network-based vulnerability scanning. Network-based vulnerability scanners are designed to look at open ports on a computer and to attempt to see if any known vulnerabilities or exploitable services exist in the services that have opened the port. A brief aside concerning the advisability of utilizing this tool – network-based vulnerability scanning should not be used against a system in production as it may cause the system to crash. Running a vulnerability scanner against a system without specific permission (preferably written) to do so could be a career damaging move. Aside from the concern of having permission and being careful when and where to run vulnerability scans, there is the issue of what return is realized from the exercise. Network-based vulnerability scanners have been demonstrated to be less than effective in determining a system's true vulnerabilities. Network Computing<sup>1</sup> performed a test of several vulnerability scanners against five operating systems with a total of 17 known vulnerabilities. None of the tested scanners found all 17 vulnerabilities. If these tools cannot find well known vulnerabilities how comfortable should a security practitioner be that a system is secure because nothing was found in a scan? These tools may have a place in the security practitioner's toolbox, but they should not be relied upon as a definitive source of assessment.

In order to create a reproducible model or process for building a server on a particular operating system, it is essential to have a good understanding of the operating system. There is no substitute for digging in and identifying the installed components and verifying that they are running as needed and have been secured and/or patched. Many services install with default values or sample files that are well known and easily exploited. One example is the SNMP service for Windows NT 4.0. Prior to Service Pack 4, SNMP access could not be set to read only<sup>2</sup>. The default community name is "public." In addition, by default, permissions on the Windows Registry settings for SNMP are incorrect<sup>3</sup>. These issues apply to a service that is optional to install. However, in an environment which requires SNMP for network management these issues need to be recognized and dealt with.

## **Run Only the Required Services**

One way to decrease the number of patches it is necessary to deploy on an ongoing basis to an organization's servers is to eliminate the service that causes the vulnerability. Services that open network ports are potential vulnerabilities. There should not be any services on a server that are not examined for the potential to introduce vulnerability. Just as user accounts should be created with the minimum required access, servers should be built with the minimum

required services. In order to determine what services are required on a server it is necessary to know what role(s) the server will fulfill. The steps required to secure a web server would be quite different from those required to secure a file server or a print server. Through clearly defining the role of a server, it is possible to determine exactly what services are required to fulfill that role. Certain services, such as TCP/IP networking, may be required for all servers in an organization's environment. Other services, such as a web server, may be needed only on a few systems. After determining which services are required for all servers, it is possible to document decisions made regarding those services and the rationale for such decisions. This is the start of a uniform build, or model.

### **Take Advantage of Published Advisories**

Creating a secure server build requires a number of decisions as the build progresses. A systems administrator tasked with creating a secure build should take advantage of the wealth of information available regarding securing systems. If other resources are available within the organization the administrator should involve them where possible. Local, regional, or national user groups may also have resources available or contacts for sharing information. Many resources are made freely available on the Internet as well.

CERT/CC (The CERT® Coordination Center at Carnegie Mellon University) <sup>4</sup> provides one central resource for information regarding security advisories. The web site for CERT/CC also has a number of articles available on the topic of improving security<sup>5</sup> and for securing specific operating systems such as Windows NT<sup>6</sup>. This document provides links to multiple resources as well as making information available that will be invaluable in recognizing what services are running on a system and what ports the services will open. The improving security section of this web site includes issue specific modules<sup>7</sup> that can be read and reviewed in assistance of build process decision making.

SANS<sup>8</sup> provides a number of step by step guides for securing operating systems. As of the time this paper was written consensus guides were available for Solaris, Windows NT, and Linux. There are also topical guides that do not address a specific operating system. These resources are another excellent way to determine what security practitioners around the world are recommending.

While the resources available at the two listed sites are comprehensive, they are by no means the only available references. Performing an Internet search for Windows NT security or Linux security will demonstrate the wealth of information available from organizations that have determined to provide assistance in the realm of operating system security.

Vendors often provide information on building their operating system securely. Microsoft has advice documents available for securing Windows NT<sup>9</sup>. There are

also numerous checklists on Microsoft's web site that can be followed to secure several of their products<sup>10</sup>. While these documents can sometimes be difficult to find they are certainly worth reviewing. The important point to remember is that there are resources available to help build a secure model for most current operating systems.

In all cases, the system administrator, or the team developing the build documentation, must make informed decisions as to which recommendations to apply. Certain of the recommendations may be determined to remove required functionality from a server. Others may be found to be unnecessary because the service involved has been left off or is disabled. It is then essential to make a risk assessment and determine whether the recommended action should be taken. If a system must perform functions that would be removed by following recommendations there should be a note in the build documentation as to what suggested action was not taken and why the decision was made.

### **Document Decisions**

As a standard build progresses, decisions will be made on installation of services or configuration of the server. Each step should involve a test to establish that the required change has not caused the server to be unable to perform its intended function. After testing has shown that a configuration choice is appropriate, the process involved should be documented. The rationale for the decision should be documented as well. That way there is not a situation one day where no one remembers why it was necessary to load the SNMP service. Also, if the SNMP service was loaded for a specific reason and that reason is no longer valid, the service can be removed after appropriate change management procedures have been followed. The specific steps involved in installing a component must be documented in order to ensure that the process can be duplicated. It is often helpful to have a second person standing by to write down any deviation from the standard so that the build documentation does not rely on one person's memory. In some organizations there will be formal procedures to follow for documentation. They may even maintain a document, or set of documents, entitled the "continuity book." This term refers to a tool by which the person leaving a position can pass on relevant information to the new person entering the position. Where possible, the model documentation should be reviewed by another person. Ideally, another technician would utilize the document to build an identical server, thus demonstrating that the secure build is reproducible.

Documenting each step will help in accrediting new servers. The document can be used as an algorithm for the server builds. At the time it becomes necessary to reaccredit a server, the build document can be utilized to see if the services and settings are still in compliance with the standard. It is worth noting that there are tools available to check servers against standards, but that it is still a requirement to know what should be on the server. There is no substitute for

having organizational knowledge of the components selected for a build and the rationale for selection. Organizational knowledge cannot be assured through staff training. Organizational knowledge is retained by means of documentation.

A server build document that adequately describes the steps required to bring up a secure server can serve as documentation for business continuance. If an organization is serious in the intent to provide business continuity (or disaster recovery), the ability to replace and rebuild servers should be included as a planning component.

### **Review and Update as Appropriate**

As time passes, the decisions made in building a server may no longer be valid. New vulnerabilities will be disclosed or new patches made available. A server may also be providing a service that is no longer needed or be required to perform an additional service. If a configuration change or patch is deemed necessary, the build document needs to be updated to reflect the change. Once again, the object is to identify all actions that brought a system to its current state in order to ensure that the current state can be reproduced.

The documentation of a secure build should be reviewed on a periodic basis in order to ensure continued viability of the build. A system that was built one year ago may have had numerous vulnerabilities disclosed regarding operating system components or installed software. A commitment must be made to review and refresh the build document on a scheduled basis. The schedule for reviewing a build model will vary between organizations. This is often determined by the available staff or contractor resources and anticipated workload.

It is advisable to assign responsibility within an organization to monitor announcements of vulnerabilities as they are made. The individual(s) involved should have the resources to help determine if an advisory applies to systems within the organization. One tool that can be utilized in this function is the build model. If the build model adequately documents the system components and version, it will be easier to determine if a specific issue applies. If a disclosed vulnerability is determined to apply to deployed servers, a process should exist in support of testing and deploying available patches or configuration changes. These changes must be reflected in the model documentation as well.

### **Utilize Change Management**

Changes to systems usually impact a number of people. Prior to making a change to a server, an administrator should submit the proposed change for review. Different organizations will have different change or configuration management requirements. In order to maintain a secure model, an organization should require review of changes to the servers built under the

model. Undocumented changes represent two risks. Firstly, the change may cause decisions that were made in building the server to be invalidated. An example of such a change is the decision to install print services on a server which did not previously provide them. Under UNIX, various printing packages have been demonstrated to have significant vulnerabilities over time.<sup>11</sup> Adding print services to a UNIX (or other) server without examining the potential risk could invalidate the accreditation of a secure server. The second risk is that an undocumented change will not be implemented on new systems or replaced in the event that a system must be rebuilt. If possible, changes should be applied against a non-production server. The non-production server should then be rigorously tested in order to assure that the changes would not make production systems unavailable if applied.

## References:

---

<sup>1</sup> Forristal, Jeff and Shipley, Greg. "Vulnerability Assessment Scanners." Network Computing. January 8, 2001. URL: <http://www.networkcomputing.com/1201/1201f1b1.html>. (February 6, 2001).

<sup>2</sup> Windows NT SNMP Security Permissions. Network Associates Security Advisory # 30. November 17, 1998. URL: <http://www.pgp.com/research/covert/advisories/030.asp>. (February 6, 2001).

<sup>3</sup> Microsoft Security Bulletin (MS00-095). December 6, 2000. URL: <http://www.microsoft.com/technet/security/bulletin/ms00-095.asp>. (February 6, 2001).

<sup>4</sup> CERT Coordination Center home page. January 31, 2001. URL: <http://www.cert.org/>. (February 7, 2001).

<sup>5</sup> Improving Security page. CERT Coordination Center. October 10, 2000. URL: <http://www.cert.org/nav/securityimprovement.html>. (February 6, 2001).

<sup>6</sup> CERT Coordination Center. Windows NT Configuration Guidelines. April 17, 2000. URL: [http://www.cert.org/tech\\_tips/win\\_configuration\\_guidelines.html](http://www.cert.org/tech_tips/win_configuration_guidelines.html). (February 6, 2001).

<sup>7</sup> CERT Security Improvement Modules. CERT Coordination Center. July 12, 2000. URL: <http://www.cert.org/security-improvement>. (February 7, 2001).

<sup>8</sup> SANS Institute Online home page. URL: <http://www.sans.org>. (February 7, 2001).

<sup>9</sup> Secure\_NTInstall. Microsoft NT Server Security Services pages. June 05, 2000. URL:

---

[http://www.microsoft.com/NTServer/security/exec/overview/Secure\\_NTInstall.asp](http://www.microsoft.com/NTServer/security/exec/overview/Secure_NTInstall.asp). (February 6, 2001).

<sup>10</sup> Security Tools. Microsoft TechNet web pages. January 3, 2001. URL: <http://www.microsoft.com/technet/security/tools.asp>. (February 7, 2001)

<sup>11</sup> Below are two representative examples of UNIX print service vulnerabilities:

CERT® Advisory CA-1997-19 lpr Buffer Overrun Vulnerability. CERT Coordination Center. April 7, 1998. URL: <http://ciac.llnl.gov/ciac/bulletins/h-74.shtml>. (February 6, 2001).

CERT® Advisory CA-2000-22 Input Validation Problems in LPRng. . CERT Coordination Center. December 12, 2000. URL: <http://www.cert.org/advisories/CA-2000-22.html>. (February 6, 2001).

© SANS Institute 2000 - 2005, Author retains full rights.



# Upcoming Training

Click Here to  
**{Get CERTIFIED!}**



SANS vLive - SEC401: Security Essentials Bootcamp Style	SEC401 - 201710,	Oct 23, 2017 - Nov 29, 2017	vLive
SANS San Diego 2017	San Diego, CA	Oct 30, 2017 - Nov 04, 2017	Live Event
San Diego Fall 2017 - SEC401: Security Essentials Bootcamp Style	San Diego, CA	Oct 30, 2017 - Nov 04, 2017	vLive
SANS Seattle 2017	Seattle, WA	Oct 30, 2017 - Nov 04, 2017	Live Event
SANS Gulf Region 2017	Dubai, United Arab Emirates	Nov 04, 2017 - Nov 16, 2017	Live Event
Community SANS Vancouver SEC401^	Vancouver, BC	Nov 06, 2017 - Nov 11, 2017	Community SANS
Community SANS Colorado Springs SEC401~	Colorado Springs, CO	Nov 06, 2017 - Nov 11, 2017	Community SANS
SANS Miami 2017	Miami, FL	Nov 06, 2017 - Nov 11, 2017	Live Event
SANS Sydney 2017	Sydney, Australia	Nov 13, 2017 - Nov 25, 2017	Live Event
SANS Paris November 2017	Paris, France	Nov 13, 2017 - Nov 18, 2017	Live Event
SANS San Francisco Winter 2017	San Francisco, CA	Nov 27, 2017 - Dec 02, 2017	Live Event
Community SANS St. Louis SEC401	St Louis, MO	Nov 27, 2017 - Dec 02, 2017	Community SANS
SANS London November 2017	London, United Kingdom	Nov 27, 2017 - Dec 02, 2017	Live Event
Community SANS Portland SEC401	Portland, OR	Nov 27, 2017 - Dec 02, 2017	Community SANS
SANS Khobar 2017	Khobar, Saudi Arabia	Dec 02, 2017 - Dec 07, 2017	Live Event
SANS Austin Winter 2017	Austin, TX	Dec 04, 2017 - Dec 09, 2017	Live Event
SANS Munich December 2017	Munich, Germany	Dec 04, 2017 - Dec 09, 2017	Live Event
Community SANS Ottawa SEC401	Ottawa, ON	Dec 04, 2017 - Dec 09, 2017	Community SANS
SANS vLive - SEC401: Security Essentials Bootcamp Style	SEC401 - 201712,	Dec 11, 2017 - Jan 24, 2018	vLive
SANS Bangalore 2017	Bangalore, India	Dec 11, 2017 - Dec 16, 2017	Live Event
SANS Cyber Defense Initiative 2017	Washington, DC	Dec 12, 2017 - Dec 19, 2017	Live Event
SANS Cyber Defense Initiative 2017 - SEC401: Security Essentials Bootcamp Style	Washington, DC	Dec 14, 2017 - Dec 19, 2017	vLive
Community SANS Nashville SEC401^	Nashville, TN	Jan 08, 2018 - Jan 13, 2018	Community SANS
Community SANS Hawaii SEC401	Honolulu, HI	Jan 08, 2018 - Jan 13, 2018	Community SANS
SANS Security East 2018	New Orleans, LA	Jan 08, 2018 - Jan 13, 2018	Live Event
Mentor Session - SEC401	Memphis, TN	Jan 09, 2018 - Mar 13, 2018	Mentor
Northern VA Winter - Reston 2018	Reston, VA	Jan 15, 2018 - Jan 20, 2018	Live Event
SANS Amsterdam January 2018	Amsterdam, Netherlands	Jan 15, 2018 - Jan 20, 2018	Live Event
Mentor Session - SEC401	Minneapolis, MN	Jan 16, 2018 - Feb 27, 2018	Mentor
SANS Las Vegas 2018	Las Vegas, NV	Jan 28, 2018 - Feb 02, 2018	Live Event
Las Vegas 2018 - SEC401: Security Essentials Bootcamp Style	Las Vegas, NV	Jan 28, 2018 - Feb 02, 2018	vLive