

A hand holding a key against a background of binary code. The key is the central focus, held in a hand that is partially visible on the left. The background is a dark grey with a pattern of white binary digits (0s and 1s) scattered across it. The text is overlaid on the upper left portion of the image.

*Ernst & Young LLP*  
*Security & Technology Solutions*  
Advanced Security Center

Using Attack Surface Area  
And Relative Attack Surface Quotient  
To Identify Attackability

Customer Information Paper

## TABLE OF CONTENTS

<b>Executive Summary</b> .....	<b>1</b>
<b>Relative Attack Surface Quotient</b> .....	<b>2</b>
<b>Evaluation Methodology</b> .....	<b>3</b>
<b>Analysis and Recommendations</b> .....	<b>3</b>
<b>Platform Tests and Results</b> .....	<b>4</b>
<b>Conclusion</b> .....	<b>5</b>
<b>Appendix A</b> .....	<b>6</b>

# Executive Summary

## Introduction

---

In March 2003, Microsoft engaged the Security and Technology Solutions practice of Ernst & Young LLP to validate the Relative Attack Surface Quotient (RASQ) model developed by Microsoft, which quantifies the relative “attackability” provided by each of its operating system platforms. The model provides a methodology to compute the attackability of Microsoft Windows server operating system platforms by describing potential exploit points and assigning a relative vulnerability level based on exploits that occur in the real world. In addition to a review of the model, E&Y also tested this model against specific configurations of these separating systems to obtain RASQ ranking.

This document describes the methodology used for validating the model, analysis, comments and test results from the model’s application.

## Conclusion

---

The Ernst & Young team concluded that in comparison to other Microsoft technology-based operating systems, the Relative Attack Surface Quotient of Windows Server 2003 was the lowest based on the root attack vectors described in this paper, and was thus the least attackable operating system among the Microsoft Windows-based operating systems.



## Relative Attack Surface Quotient

---

The security of a company's data and its networks is highly dependent on the underlying architecture and the technology used in implementing it. Today's companies need to look at the security features offered by various technology vendors, and need to understand the impact of those features in protecting their data.

Traditionally, quantifying information technology security elements and risk exposure has been a difficult exercise, due to the number of intangible factors involved, and lack of common standards and vendor support. Microsoft's proposed RASQ attempts to address at least one component of the security assessment process by attempting to mathematically quantify the relative attackability of IT assets.

The RASQ of a product is calculated by adding together the effective attack surface value for all root attack vectors. A root attack vector is a particular feature of an operating system that can positively or negatively affect the security of the product. The effective attack surface value is defined as the product of the number of attack surfaces within a root attack vector and the attack bias. An attack bias is the value representing the risk of compromise for an attack vector weight given to a particular attack vector that represents the threat of compromise of that attack vector.

Assumptions might be made that the attackable surfaces are potential weak spots within a product, such as network services running by default, weakly protected accounts and files, or improperly written code. However, an attackable surface does not necessarily mean a surface is vulnerable. For purposes of this model, an attackable surface is defined as a target for a potential attacker. The second component of the model, attack bias, takes the vulnerability of these attack surfaces into account, as potential attackers would likely target known vulnerabilities.

Once the root attack vectors are defined, the number of attack surfaces can be precisely calculated using well-defined processes. However, calculating the attack bias value for this review was a subjective process that assigned a value between 0 (no threat) and 1 (maximum threat) based on the particular attack vector's threat and its attractiveness to a potential attacker. Ideally, the bias values should be calculated based on the threat level for each component covered by a single attack vector, as well as the particular role/function of the system.

For example, if the number of default services running under the system account was the attack vector, the scheduler service running under the system account on a domain controller may be considered to provide a higher level of threat than the same service on a stand-alone server. This is an even lower level of threat for a different service, such as the IPSEC policy agent, even if it is started as default and runs under privileged accounts.

In short, the credibility of RASQ scores depends on the business environment and the specific role of the system to be tested.



## Evaluation Methodology

---

### *The Team*

The E&Y assessment team utilized their extensive experience in researching, designing, testing and building security solutions to objectively assess the RASQ model, and test the OSs. The primary assessment team also utilized E&Y experts in mathematical modeling, statistical analysis and other technologies as needed throughout the course of this engagement.

### *The Environment*

All of the product tests were conducted at the Advanced Security Center in Houston. The products evaluated were installed on Intel Pentium systems using default installation options and as workgroup members on a test network. Widely available tools such as nmap, netstat, pipelist, rpcdump, pulist, fport, SCM and more were used to measure the attack surface values whenever possible. Further, custom scripts were developed and used during this assessment.

### *RASQ Validation*

Due to the lack of existence of a similar model that compares attackability, E&Y compared RASQ with other quantitative security methods such as the NIST and Holistic models, among others. These comparative analyses, combined with the team's experience and security knowledge base, was used during the validation exercise to arrive at the bias values used during the product tests and other recommendations in this white paper.

## Analysis and Recommendations

---

### *RASQ Model*

As a practical quantitative approach, the RASQ model can meet its stated objective of quantifying the attackability of an operating system provided the following conditions are understood:

- The RASQ model's objective should not be misinterpreted to measure a system's vulnerability to attacks, and/or more importantly its security risk.
- Test results based on this model are only meaningful for products of a similar nature. This is an especially important condition because the root attack vectors used by the model are very dependent on specific technologies and features.
- Since the model uses parameters that are only valid within the Microsoft domain of operating systems, its applicability at this stage should be restricted to Microsoft platforms.
- The RASQ scores alone may not signify that a platform with a lower RASQ score is more secure than a platform with a higher RASQ score.

### *Root Attack Vectors and Attack Bias*

The root attack vectors used in this assessment were based on default product configurations, which sometimes included services or features that might be considered optional or undesirable in an actual system. Ideally, the root attack vectors used by the model should be the minimum base features of the products evaluated and not necessarily what is packaged with the product. Optional attack vectors could then be defined based on the role of the system or platform under review. For example, metrics related to the system's role as an IIS, SQL or Active Directory Server (e.g. default object permissions) or Domain Controllers (e.g.: authentication credentials, storage of authentication credentials, etc.) Please refer to Appendix A for understanding the root vectors and bias values used.



## Platform Tests and Results

Using the root attack vectors defined above, E&Y tested the RASQ model on different versions of Microsoft operating systems with different service packs and optional feature combinations. The results in Table 1 show the RASQ values for each tested platform.

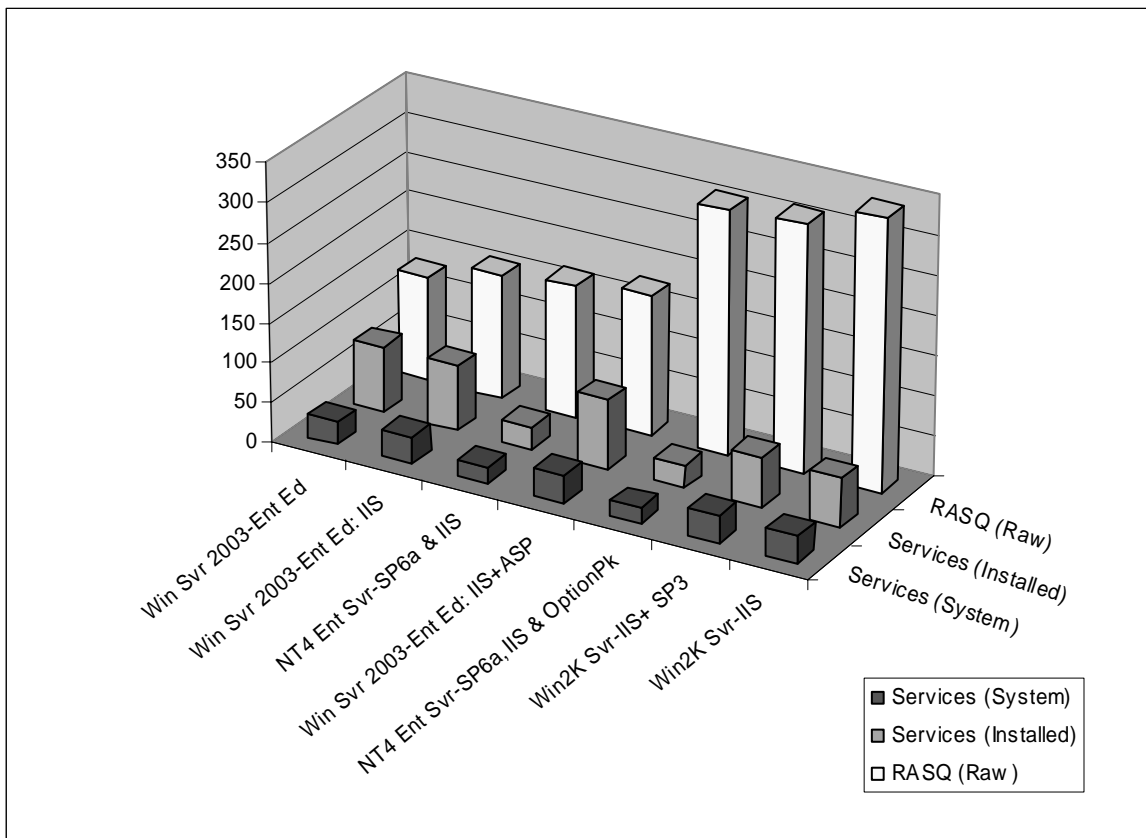
The current version of Microsoft's Workstation OS was also tested as part of the model validation exercise. Enabling the Internet Connection Firewall (ICF) did lower the RASQ score for that platform, demonstrating the model's functionality.

Among the server versions of the operating systems, NT4 without IIS obtained the lowest RASQ score. This can be attributed to IIS related metrics significantly affecting the RASQ scores.

Product Configuration	RASQ
Windows Server 2003-Ent Ed	131.30
Windows Server 2003-Ent Ed: IIS	156.60
NT4 Ent Server-SP6a & IIS	166.10
Windows Server 2003-Ent Ed: IIS+ASP	177.80
NT4 Ent Server-SP6a, IIS & OptionPk	307.00
Win2K Server-IIS+ SP3	311.90
Win2K Server-OOB w/IIS	341.20

**Table 1: RASQ Scores for Tested Platforms**

It is interesting to note that Microsoft has addressed several of the security shortfalls in its server operating systems with the introduction of its Windows Server 2003 platform. It has the lowest RASQ score among the server OSs configured with IIS. A significant reduction in the number of dynamic web pages and active web handlers that are installed by default has improved its RASQ score compared with the Windows 2000 server.



Windows NT4's excessive number of files with weak access permissions, along with the number of dynamic web pages installed by default, contributed to its much higher RASQ score, as expected.

However, Windows Server 2003 does have a higher number of open RPC endpoints, named pipes and Services (Installed – not running) than either Windows NT or Windows 2000. However, since these attack vectors had a lower bias value, due to the lower threat level posed by these vectors, the RASQ scores were not significantly affected.

## Conclusion

---

The results of E&Y's assessment of the RASQ model and the usage of this model in conducting a set of tests against representative systems confirmed that the model is useful in quantifying the relative differences in security between similar operating systems. In addition, this model correlates with the team's real world experience in conducting security assessments, and the assumption that a model of lesser complexity creates less chance of errors or security flaws.

The testing of the representative operating systems also indicated the relative improvements in security Microsoft has made with the Windows Server 2003. Security issues around IIS server's default/sample content seems to have been a priority for Microsoft with this version of its server OS, Windows Server 2003. Logical separation of the system account and the reduced level of privileges offered by the new system account variants offers a higher degree of security against privilege escalation type attacks.

While security of information systems is dependent on many factors, the use of a formal model such as Microsoft's RASQ appears to be a reasonable indicator of the security of Microsoft's operating systems. The use of the model should be understood to be limited to a specific set of conditions that are controlled by the appropriate type of technologies, and also in the sense that it does not directly predict the security risk of the platform. However, this model demonstrates reasonable security principles, and, furthermore provides significant additional metrics that can be used as design principles for developing more secure systems, as proved in the release of its latest server operating system, Windows Server 2003. Application of this model and its basic security principles may allow even greater reduction of the attack surface, likely resulting in increased security of future versions.



## Appendix A

---

For this product evaluation, the following root vectors and bias values were used:

1. **Open Sockets:** A bias of 1.0 was used, as every open and listening socket could be used as a potential target by attackers.
2. **Open RPC Endpoints:** A bias of 0.9 was used, as every open RPC endpoint is a potential attack target. However, RPC exploits usually need higher skill levels than open sockets.
3. **Open Named Pipes:** This vector was assigned a bias of 0.5 because of the attack complexity required, as well as its practical exploit potential.
4. **Services:** These were given a bias of 0.2, because not every service is running and/or is potentially vulnerable. However, the local admin, or a potential attacker who has managed to compromise the local admin account, could still start the services to attempt to further elevate privileges.
5. **Services Running by Default:** A bias of 0.8 was assigned, as every running service that is remotely accessible and/or controllable can be considered a potential attack surface.
6. **Services Running as System:** A bias of 0.9 was used as services that run under the system context provide a more enticing target to an attacker, since these services, if successfully exploited, could grant elevated access privileges to the attacker.
7. **Active Web Handlers:** This was given a bias of 1.0, because each active web handler is a potential attack target.
8. **Active ISAPI Filters:** This was assigned a bias of 1.0, as each active filter is a potential attack target.
9. **Dynamic Web Pages:** A bias of 0.6 was assigned because each dynamic web page could contain code that could make it vulnerable to an attack like cross-site scripting. However, since every default dynamic web page is exploitable, a slightly above median threat was assigned.
10. **Executable Virtual Directories:** A bias of 1.0 was assigned to this vector, as every directory with execute permissions set could let an attacker run scripts or executables in that directory, combined with other vulnerabilities.
11. **Enabled Accounts:** Default accounts simplify “password brute-force” type attacks, as these account names are widely known. However, standard security measures such as password protection are available to prevent easy access into the system. Because of this, this vector was assigned a bias of 0.7.
12. **Enabled Accounts in the Administrator Group:** Every account in the administrator group is a higher risk than other normal accounts due to the widespread knowledge of these account names and powered offered by them, thus an increased bias of 0.9 was used.
13. **Null Session access to Pipes and Shares:** This was given a bias of 0.9 because every null session is a potential way to connect to a remote machine to enumerate the system environment and/or further the attack attempt.
14. **Guest Accounts Enabled:** These are assigned a bias of 0.9, as default guest accounts that are enabled often provide a potential way to connect to a remote system to enumerate the system environment and/or further the attack attempt.
15. **Weak ACLs in File System:** Since most files in the system directories are targeted after the system has been compromised, this was assigned a lower bias of 0.2.
16. **Weak ACLs in Registry:** These are given a slightly higher bias value of 0.4, as remote registry access is commonly allowed.
17. **Weak ACLs on File Shares:** These are given a bias of 0.9, as default share names are commonly known and are targeted quite often by remote attackers for various exploitation purposes.

