Enterprise Strategy Group

Information Security Brief

Microsoft SQL Server Runs the Security Table

Date: November, 2006

Author: Eric Ogren, Security Analyst

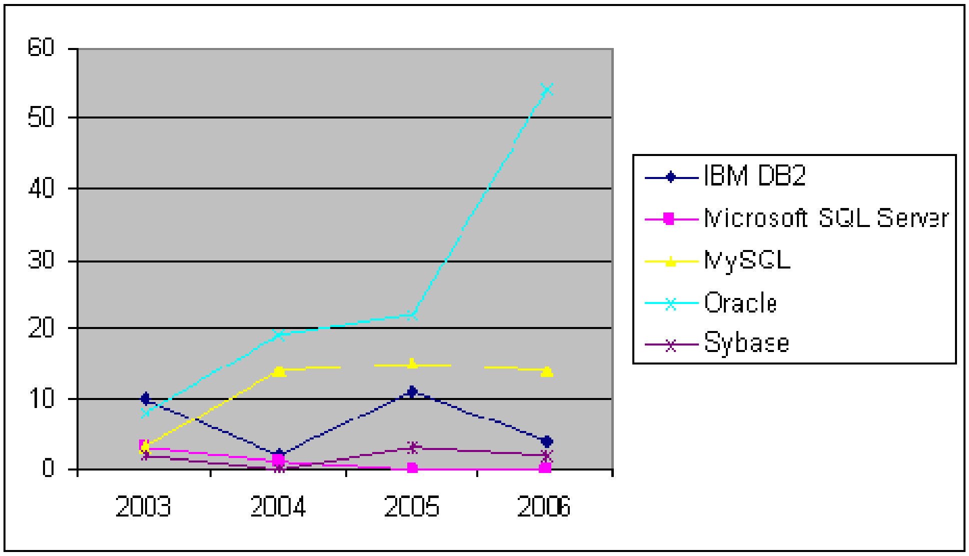
Abstract: The rate of security vulnerabilities documented in the National Vulnerability Database for the major database vendors is noteworthy for the stark contrast between Microsoft, MySQL and Oracle. ESG believes that Microsoft's investments in secure development processes are responsible for the impressive results in SQL Server quality. ESG considers Microsoft, with proper execution, to be years ahead of Oracle and MySQL in producing secure and reliable database products.

Overview

The health of databases is of critical importance to business managers, application owners and enterprise IT teams. The life of an organization is literally represented inside its database servers. Take away the ability to reliably run enterprise applications or complete customer transactions and you will watch the business come to a standstill. One quantifiable indicator of risk to the enterprise of business disruption, or leakage of confidential data, is the number of vulnerabilities that exist in the technical infrastructure. There is a correlation between the number of vulnerabilities and the number of undiscovered vulnerabilities as well as the risk to the enterprise of an exploit successfully launched against the vulnerable database. It is very clear that the more vulnerabilities that exist, the more likely it is that an attack will be successful.

With this in mind, ESG compiled Common Vulnerabilities and Exposures (CVE) data from the National Vulnerability Database to compare security vulnerabilities between commercial database offerings from Microsoft, Oracle and the open source MySQL. Oracle, traditionally a company that holds its security cards very close to the vest, has been disclosing large numbers of security vulnerabilities over the past few quarters. Microsoft has gone through its own soul-searching when it comes to security, from openly disclosing defects to a complete revamping of its engineering process with its Security Development Lifecycle (SDL). Microsoft SQL Server 2005 is particularly worth examining as it is the first major product release that Microsoft has put through the SDL.

Figure One: Common Vulnerabilities and Exposures, 2003-2006
Source: Compiled from National Vulnerability Database, November, 2006



ESG chose to compare the number of CVE entries maintained by the National Institute of Science and Technology (NIST) because they are the best independent registry of problems discovered by customers, researchers and vendors, without the possible vendor skews that can occur when counting issued patches or security bulletins.

CVE entries have tended to increase since the year 2003, due to greater product complexity, larger installed bases using the product in different ways and an active role by NIST to gather and communicate defect information to customers. The divergence in CVE submissions amongst database vendors is noteworthy:

- Microsoft SQL Server, since launching the SDL program, has no current CVEs for 2006. This trend is maintained even as unit volumes increase.[1]
- MySQL, the open-source database offering, stands at 14 current CVE entries thus far in 2006. [2]
- Oracle Database has jumped to 54 current CVE submissions thus far in 2006.[3]
- ESG did not include IBM DB2 and Sybase since the security discussions have centered around Microsoft, Oracle and the potential of open source approaches. For the record, IBM DB2 has 4 CVE entries in 2006[4] and Sybase has 2.[5]

Oracle's results over the past two years show that much work has to be done to bring the vulnerabilities into line with competing database products from IBM, Microsoft, MySQL and Sybase. While Oracle is to be commended for the candor and thoroughness of its disclosure program, it will take a significant period of time for Oracle engineering to get to the root causes of the high rate of vulnerabilities and to implement corrective development procedures. ESG believes that there are no shortcuts in this process.

Microsoft's results are almost too good to believe, and thus serve as a model for other database vendors. ESG expects the CVE trend to continue through 2007 when changes in development approaches by Oracle and MySQL will start to yield more secure software in customer deployments.

What has Microsoft done that the industry can learn from?

ESG finds that Microsoft has made significant investments in improving the security and integrity of its products. The results of those investments are impressive, as shown in Figure One. ESG believes that the other database vendors can benefit from the Microsoft experience and learn from the best practices of Microsoft development:

- **Bake security into the core.** Security is not a feature that can be sprinkled onto enterprise application systems. It has to be a measured objective from Day One. Microsoft makes every effort to catch defects before they become embedded in designs and source code with mandatory security training of engineering staff, peer review of source code, formal sign-offs of security reviews and rigorous forensic analysis of discovered defects to be sure the organization learns how the defect could have been prevented.
- **Reduce attack surfaces.** Active interfaces provide entry points for attacker exploits, usually by finding a weakness in parameter handling. Reducing the attack surface provides software products that are easier for Microsoft and its customers to secure. While it is convenient to have all product features tacitly enabled by default, good security practice dictates an "opt-in" approach. Microsoft products will ship in secure configurations where the customer will explicitly enable desirable product features.
- **Break the implementation before customers do**. This attitude goes way beyond traditional Quality Assurance functions. Engineers, who know the product best, are required to complete a process of Threat Modeling followed by a series of Fuzz Tests. Threat Modeling uses the top engineers to focus

[1] http://nvd.nist.gov/, advanced search of Microsoft SQL Server, Microsoft SQL Server Desktop Engine, SQL Server Desktop Engine

[2] http://nvd.nist.gov/, advanced search of MaxDB, MySQL

[3] http://nvd.nist.gov/, advanced search of 9i, Oracle 9i, Oracle 9i Release 1 and 2, Oracle 9i Database Release 2, Oracle 9i Database Server, Oracle 9i Database Server Release 1 and 2, Oracle 9i Lite, Oracle 10g, Oracle 10g Database Server, Oracle Database Server Release 1 and 2, Oracle 10g Enterprise Manager Database Control, Oracle 10g Enterprise Manager Grid Control, Oracle 10g Reports Server

[4] http://nvd.nist.gov/, advanced search of DB2, DB2 Content Manager, DB2 Universal Database, DB2 Universal Database for AIX. HP-UX, Linux, Solaris, and Windows

[5] http://nvd.nist.gov/, advanced search of Adaptive Server Anywhere, Adaptive Server Enterprise, EA Server

on design and architectural linkages to uncover threat scenarios and solutions. Fuzz Tests automate penetration testing of the attack surface with distorted parameters to detect vulnerabilities.

The Microsoft approach is to make building secure products intrinsic to the entire development process. While they may use tools to automate detection of such things as the use of banned routines or insecure coding practices, the real value is in creating awareness, providing education and increasing team responsiveness throughout the organization. It is a tremendous investment in resources that has taken years in which to achieve demonstrable results in customer deployments.

SQL Server 2005 implements SDL

Microsoft chose to use SQL Server as the first product to commit to the SDL process, starting with SQL Server 2000 SP3 released in 2003 and continuing with SQL Server 2005. The results, as measured by the CVE entries of Microsoft's closest competitors have been impressive. While Oracle has shown steady year-on-year growth in registered defects, the National Vulnerability Database reports zero CVEs for Microsoft SQL Server in both 2005 and 2006.

- **Secure by Design.** SQL Server has made many changes to bake security into the database. SQL Server implemented some of the outcomes of Threat Modeling exercises, such as reducing the risk of unintended database access by requiring administrators to explicitly enable ownership chaining across database boundaries or implementing a granular trust model so well-written applications can avoid the security problem of hardcoding usernames and passwords in application code.
- **Secure by Default.** Consistent with reducing attack surfaces, SQL Server disables many options by default. Customers installing SQL Server out of the box start with a secure configuration where they have to explicitly enable required services. For instance, customers must enable features that are turned off by default, such as Windows command shells, dbmail and SQL browser service. The chance of an attacker taking advantage of inadvertently open external connections is significantly reduced.
- **Secure by Deployment.** Microsoft has implemented features in SQL Server that make it a more secure database product for customers to configure. ESG found the use of authenticated identity to be particularly useful in managing permissions for database actions. In fact, users need specific privileges to even be able to have visibility into metadata that they do not own.

The Bottom Line

The CVE numbers don't lie. The noteworthy results of Microsoft's investments to produce more secure software in SQL Server 2005 are a matter of public record. ESG has talked with customers that have standardized their mission critical applications on Microsoft SQL Server based on security and reliability results. The nature of the security and reliability improvements, namely fundamental changes in the way software is designed, built and tested creates an advantage that Microsoft should be able to sustain with proper execution. ESG considers Microsoft to be years ahead of Oracle and MySQL in producing secure and reliable database products.