

## Upgrading Quest® Toad® for Greater Security

A technical brief on upgrading Toad products for the software security standards built into them

By Julie Hyman, Senior Product Manager, and John Pocknell, Senior Market Strategist, Quest Software



“We’re not installing any vendor’s software unless they can demonstrate that it’s secure. There’s too much at risk.”

IT administrators are right to draw a line in the sand when it comes to the software they install – and allow their users to install – on their desktops and network. Their strictest imperatives are to ensure that:

- only secure software is installed, and
- the software is continually up to date to keep it secure.

In this technical brief you’ll find support for both of those imperatives. You’ll see information that Quest® maintains internally to document its security controls. You’ll find reassuring detail about Quest’s adherence to software security standards for continually testing and delivering versions of Toad products that you can safely install and run. You’ll also see Quest’s compelling argument for consistently upgrading Toad products so you can avoid vulnerabilities and security threats.

### DATA SECURITY AND TOAD

First, think about security as it relates to your company’s databases, which your database administrators, developers and analysts use Toad to explore. In an age of data breaches and privacy concerns, IT administrators are right to ask the question, “Where does our data go when we use Toad?”

It goes where you store it. Nowhere else.

Toad products are on-premises software applications. As such, you are in full control of the data and of the security measures in place to protect it. Toad is not a hosted or SaaS solution and, therefore, does not transmit, store, access or process your data beyond its location.

### APPLICATION SECURITY AND UPGRADING TOAD

Next, think about your role in keeping the network secure by continually updating your software.

“Where does our data go when we use Toad?”  
It goes where you store it. Nowhere else.

The ransomware attacks allowed by the WannaCry/Petya vulnerability demonstrated that hundreds of thousands of computers worldwide were running outdated software. (Many of them probably still are.) Although the fix for WannaCry was distributed as part of a normal update to Windows, too many companies did not apply the update. Despite months of warnings, the Petya attack was a second wave against the same vulnerability, affecting computers that had not yet been updated.

Granted, some IT groups delay updates that force a system restart and interrupt productivity. Some worry about how updates will affect applications upon which thousands of their users depend. Others give priority only to necessary updates and make their own call as to

what “necessary” means. Still others have an “ain’t-broke-don’t-fix-it” attitude towards upgrading Toad and other development software.

Many Toad customers have found themselves in that last category, asking the question, “Toad is a fantastic product and the version we have now runs fine on hundreds of our desktops. Why should we bother with upgrading Toad?”

In that regard, Quest is something of a victim of its own success. Although versions of Toad written years ago are still compatible with current database versions and operating systems, Quest has continually released newer, more secure versions. As an example, Table 1 lists the partial release history for Toad for Oracle.

RELEASE	GENERAL AVAILABILITY DATE
Toad for Oracle v14.0	Planned Q4 2020
Toad for Oracle 13.4	14-Aug-2020
Toad for Oracle 13.3	05-Mar-2020
Toad for Oracle 13.2	18-Nov-2019
Toad for Oracle 13.1.1	14-May-2019
Toad for Oracle 13.1	18-Oct-2018
Toad for Oracle 13.0	03-May-2018
Toad for Oracle 12.12	12-Oct-2017
Toad for Oracle 12.11	08-Jun-2017
Toad for Oracle 12.10	29-Sep-2016
<i>Toad for Oracle 12.9</i>	<i>16-Jun-2016</i>
<i>Toad for Oracle 12.8</i>	<i>03-Nov-2015</i>
<i>Toad for Oracle 12.7.1</i>	<i>27-Aug-2015</i>
<i>Toad for Oracle 12.7</i>	<i>04-Jun-2015</i>
<i>Toad for Oracle 12.6</i>	<i>25-Sep-2014</i>
<i>Toad for Oracle 12.5.1</i>	<i>10-Jul-2014</i>
<i>Toad for Oracle 12.5</i>	<i>05-Jun-2014</i>
<i>Toad for Oracle 12.1.1</i>	<i>10-Feb-2014</i>
<i>Toad for Oracle 12.1</i>	<i>19-Sep-2013</i>
<i>Toad for Oracle 12.0.1</i>	<i>01-Aug-2013</i>
<i>Toad for Oracle 12.0</i>	<i>20-Jun-2013</i>

Table 1: Toad for Oracle release history (*italicized versions no longer supported*)

Newer versions of Toad products, like Toad for Oracle, extend far beyond functional improvements to include substantial security improvements. Toad's evolution through our release process closely follows the ever-changing landscape of vulnerabilities and threats in database development, including:

- Insecure (unencrypted) connections to a database
- Unintended privilege escalation (e.g., low-privilege user to superuser)
- Incorrectly low barriers to accessing personally identifiable information (PII) and sensitive data stored in the cloud
- PII stored (and forgotten) in non-production copies of databases

At least twice a year, Quest's customers have the opportunity of upgrading Toad to the current version and adding security improvements that mitigate threats like those.

**SECURITY CONTROLS  
FOR PREVENTING SUPPLY  
CHAIN ATTACKS**

Most IT administrators are occupied with plugging security gaps among different applications installed on different platforms. That's why Quest believes in making security practical by releasing products that are secure in the first place.

Quest sees growing concern about supply chain attacks, in which malware is introduced before products are released to customers. A series of scanning controls is in place to follow software security standards for identifying and eliminating malware in the build process of Quest's Toad products, including Toad for Oracle. The controls ensure that products remain free of vulnerabilities and malware, do not contain hidden backdoors and are developed by employees and contractors acting with integrity. Table 2 shows the security controls which every release of Toad undergoes.

Toad's evolution closely follows the ever-changing landscape of vulnerabilities and threats in database development.










Security Control	Description
 <b>1. Security Training</b>	Developers, managers and directors are required to take 2.5 hours of assigned Security Training.
 <b>2. Secure Software Development Lifecycle</b>	Development lifecycle is based on best practices of Quest's Information and Systems Management (ISM) business unit.
 <b>3. Third-Party Software</b>	Bundled third-party software is checked for vulnerabilities in a standard process before application release.
 <b>4. Vulnerability Scanning</b>	All software is scanned for vulnerabilities with an industry-standard SAST/DAST product.
 <b>5. Penetration Testing (Third-Party)</b>	Products undergo third-party penetration testing annually.
 <b>6. Malware Scanning</b>	All products are scanned for malware before release with two independent, industry-standard anti-malware scanners.
 <b>7. Code Signing</b>	Software distributed to customers is cryptographically signed using Quest's official signing key to validate authenticity.
 <b>8. Software Integrity</b>	Checksums for software installers are published so customers can ensure integrity of distributed software.
 <b>9. FIPS Compliance</b>	Using cryptographic algorithms approved by FIPS, sensitive data is protected at rest and in transit.

Table 2: Toad security controls against supply chain attacks

The Secure Software Development Lifecycle (SSDLC) is designed to help Toad engineers write secure software, comply with software security standards and keep engineering costs low.

Following are details of how Quest Software implements each control for Toad products.

## 1. SECURITY TRAINING

All Quest software engineers, architects and managers complete two web-based training sessions delivered through LinkedIn Learning: “Developing Secure Software” (1 hour) and “Programming Foundations: Secure Coding” (1.5 hours). Quest reviews the training content annually to confirm it meets the requirements for developing secure software.

The goals of the training are to develop a baseline of security knowledge, improve secure coding practices and improve adherence to the Quest Secure Software Development Lifecycle (see below). Training topics currently covered include:

- Understanding Software Security
- Software Security Threats
- Secure Software Design
- Secure Coding
- Testing for Security
- Security and Risk Overview
- Web Client Server Interaction Code Issues
- Thick App and Client-Server Interaction Issues
- Crypto and Security Misuse Issues
- Security in the Software Development Lifecycle

The training sessions address multiple software security standards, including:

- NIST SP 800-53 R4 AT-2
- NIST SP 800-53 R4 AT-3
- ISO 27001 A.8.2.2
- PCI DSS v3.0 12.6

Quest tracks completion of each course and retains records for use during customer audits. The records demonstrate that Quest Engineering teams receive training in secure development practices.

## 2. SECURE SOFTWARE DEVELOPMENT LIFECYCLE (SSDLC)

The SSDLC for Toad products introduces security and privacy considerations to the process of developing the product itself. The SSDLC is designed to help Toad engineers write secure software, comply with software security standards and keep engineering costs low.

The components and phases in the SSDLC include:

- Define security requirements
- Define metrics and compliance reporting
- Perform threat modeling
- Define and use cryptography standards
- Manage risk of using third-party components
- Establish a standard incident response process

## 3. THIRD-PARTY SOFTWARE AND COMPONENT PATCHING

Like many vendors of enterprise software, Quest often incorporates code written by other companies (“third parties”) so that Quest does not have to recreate it from scratch. But over time, vulnerabilities that require patching turn up in some third-party components.

Besides relying on the NIST CVE Database for information on current vulnerabilities, Quest uses tools to identify third-party DLLs and any vulnerabilities that may have been identified in those DLLs. Quest has also developed a process for reducing the likelihood of releasing software with vulnerable third-party components. The process:

- Specifies criteria for no-ship vulnerabilities
- Requires an inventory of all third-party components, software and DLLs bundled into products
- Mandates regular checks for vulnerabilities in third-party components
- Establishes deadlines (30/60/180 days) for third parties to patch vulnerable components
- Mandates notices to customers if products are patched and a new release is issued

### The Case for Upgrading Toad

By performing regular checks as part of the build/release process, Quest identified and addressed a vulnerability in Toad for Oracle v13.1.1, then notified customers as follows:

“Quest recently patched one of these [third-party] libraries which contained a critical (CVSS 9.3) vulnerability that was discovered. The vulnerability affected the Microsoft Visual C++ library and could lead to a privilege escalation. This vulnerability has been remediated with our latest release of Toad.”

The process addresses multiple software security standards, including:

- [NIST SP 800-53 R4 SI-2](#)
- [ISO 27001 12.6.1](#)
- [AICPA TSC 2014 CC7.1](#)
- [PCI DSS v3.0 6.2](#)
- [HIPAA 45 C.F.R. § 164.308\(a\)\(1\)\(i\)\(A\)](#)
- [HIPAA 45 C.F.R. § 164.308\(a\)\(1\)\(i\)\(B\)](#)

### 4. VULNERABILITY SCANNING (SAST AND DAST)

Quest has defined processes around SAST (Static Application Security Testing) and DAST (Dynamic Application Security Testing) scanning, using external tools.

SAST is a form of white-box testing. A tester using SAST examines the application from the inside, searching its source code for conditions that indicate potential security vulnerabilities. DAST is a form of black-box testing, from the outside as an attacker would see the application. A tester using DAST examines a web application when it is running and tries to hack it as an attacker would.

SAST tools assist in identifying weaknesses found on a list known as the [common weakness enumeration \(CWE\)](#). The tools are limited in their handling of issues of logical flow, authentication and authorization, which are better suited to penetration tests (see below) or manual source code reviews. DAST scanners, interacting with a web application from the outside, rely on HTTP and are technology-independent.

For Toad products, the process includes the following:

- Full scans of the product/web application are performed automatically where possible, at least twice per year and before all releases, using the currently designated SAST/DAST tool.
- All product code developed by Quest is scanned by the SAST/DAST tool.
- The Toad security advocate reviews the results of each scan and, when necessary, works with the InfoSec Principal Engineer to determine the severity of any issues found.
- No products are released with critical or high vulnerabilities.
- For medium and low vulnerabilities, the security advocate works with the product architect.
- The security advocate and the product architect determine the best solution to any vulnerabilities uncovered by the scans.

The process addresses multiple software security standards, including:

- [NIST SP 800-53 R4 RA-5](#)
- [NIST SP 800-53 R4 SI-2](#)
- [AICPA TSC 2014 CC7.1](#)
- [ISO 27001 12.6.1](#)
- [PCI DSS v3.0 6.2 and 11.2](#)
- [HIPAA 45 C.F.R. § 164.308\(a\)\(1\)\(i\)\(ii\)\(A\)](#)
- [HIPAA 45 C.F.R. § 164.308\(a\)\(1\)\(i\)\(ii\)\(B\)](#)

By performing regular checks as part of the build/release process, Quest identified and addressed a vulnerability in Toad for Oracle v13.1.1, then notified customers.



SAST examines the application from the inside, searching source code for potential security vulnerabilities. DAST is a form of black-box testing, from the outside as an attacker would see the web application.

## 5. PENETRATION TESTING (THIRD-PARTY)

Penetration testing demonstrates real-world impact if a vulnerability or process weakness were to be exploited. It is designed to assess security before a bad actor strikes. A penetration test is not an automated scan of an application or its source code, but a next step after automated vulnerability scanning (see above).

For Toad products, penetration testing is conducted annually. The tests, a combination of manual and automated testing, are designed to uphold software security standards in the following areas of the product:

- Application logic
- Code injection
- Local storage
- Binary exploitation and reverse engineering
- Excessive privileges
- Unencrypted storage of sensitive information
- Unencrypted transmission of sensitive information
- Weak encryption implementations
- Weak assembly controls
- Weak GUI controls
- Weak or default passwords

### The Case for Upgrading Toad

As a result of penetration testing, Quest identified and addressed a vulnerability in Toad for Oracle v13.3, then notified customers as follows:

“During a recent penetration test an issue was discovered in which a user was not notified when connecting to a database insecurely. If a user unknowingly connects to a system over a non-encrypted link, an attacker can capture a user’s credentials or all data transferred across the connection. Quest has remediated this weakness.”

In most cases a penetration test follows a specific framework depending on the target application or infrastructure, and tactics vary depending on the adversary being mimicked. The most commonly used frameworks include:

- [MITRE ATT&CK framework](#)
- [PTES Framework](#)
- [OWASP Top 10](#)
- [SANS Top 25](#)

As an example, Figure 1 shows the results of third-party penetration tests conducted in 2020 on Toad for Oracle.



### Vulnerability Risk Definition and Criteria

The risk ratings assigned to each vulnerability are determined by averaging several aspects of the exploit and the environment, including reputation, difficulty, and criticality.

<b>CRITICAL</b>	Critical vulnerabilities pose a serious threat to an organization’s security, and should be fixed immediately. They may provide a total compromise of the target environment, or similar critical impacts.
<b>HIGH</b>	High risk vulnerabilities should be considered a top priority, just after critical risks. These are serious issues and post immediate security risk to the enterprise.
<b>MEDIUM</b>	Medium severity vulnerabilities represent a moderate risk to the environment. They may require additional context before remediation but should be remediated after critical and high risks.
<b>LOW</b>	Low severity vulnerabilities provide minimal risk to the target environment, and often theoretical in nature. Remediation of low risks is often a lower priority than other security hardening techniques.
<b>INFORMATIONAL</b>	Informational vulnerabilities have little-or-no impact to the target scope by themselves. They are included however, as they may be a risk when combined with other circumstances or technologies not currently in place. Remediation of informational items is not necessary.

Figure 1: Toad for Oracle 2020 penetration test results

## 6. MALWARE SCANNING OF SOFTWARE BUILDS

When Toad product builds are packaged for release, they are first scanned for malware following a consistent process that includes these steps:

- Install packages are hashed using the SHA-256 algorithm before scanning, and the hash must match the final, published hash on the package.
- All software builds are scanned for malware before they are released for install.
- All files packaged into an installer are first scanned for malware.
- Malware scanning is automated (command-line process or script).
- Independent malware scanners (two for Windows and two for Linux) are used, updated with the latest signatures/definitions before the scan.
- An evidence record including time, date and results is produced and permanently retained for each scan.
- The security and engineering directors must approve exceptions for any files in the package.

The process addresses multiple software security standards, including:

- AICPA TSC (SOC-2) CC5.8
- NIST SP 800-53 R3 SC-5
- NIST SP 800-53 R3 SI-3
- NIST SP 800-53 R3 SI-5
- HIPAA 45 CFR 164.308 (a)(5)(ii)(B)
- ISO 27001:2013 A.12.2.1
- PCI 1.4, 5.0

## 7, 8. CODE SIGNING AND SOFTWARE INTEGRITY

An application bearing Quest's code signing certificate is a customer's assurance that Quest created the application and that the software can be trusted. The code signing process serves the goal of ensuring authenticity by verifying the author of the software. It also ensures the integrity of the software by demonstrating that the code has not been altered since it was signed.

Code signing also plays a role in releasing updates and patches. When Quest signs an update to a Toad product with the same key used in the original application, it means that the update can be trusted; it couldn't have come from any source other than Quest. Finally, the checksums generated in code signing assure users that they have received the correct file, rather than a file that has been signed with a stolen key.

All major operating systems and web browsers support code signing to prevent the distribution of malicious code.

Quest signs all releases of Toad products using a trusted Quest key. During the build process, Quest signs every .exe and .dll file included in the installer, along with any binaries packaged with the application and the installer files themselves. Applications available for download include a SHA-256 checksum hash so that customers can verify the integrity of the file upon receipt.

The process addresses software security standards that include NIST SP 800-53 R4 SI-7(15).

## 9. PROTECTING SENSITIVE DATA THROUGH FIPS COMPLIANCE

The presence of sensitive data in your applications and databases imposes a burden of protection. Sensitive data extends to almost any kind of data you would want to prevent from falling into the wrong hands, including:

- Network credentials
- Passwords
- Social Security numbers
- Credit card information
- Personally identifiable information (PII) such as names, addresses and phone numbers
- Personal health information (PHI)
- Financial information
- Internal records
- Intellectual property

Quest signs every .exe and .dll file included in the installer, along with any binaries packaged with the application and the installer files themselves.

All Quest products comply with FIPS-approved algorithms for encryption and hashing. The current status of FIPS compliance (currently FIPS 140-2) is validated prior to each release.

Toad products protect sensitive data through cryptographic algorithms that conform to the Federal Information Processing Standards (FIPS) of the United States Government. Furthermore, they apply that protection to sensitive data both in transit and at rest.

The FIPS standards specify the best practices and requirements for cryptography-based security systems, including methods for encryption and for generating encryption keys. FIPS compliance is mandatory for all computers used for U.S. government work and extends to testing outside applications (like Toad) that will run on U.S. government computers.

All Quest products comply with FIPS-approved algorithms for encryption and hashing. The current status of FIPS compliance (currently FIPS 140-2) is validated prior to each release.

Toad products are built using one or more of the following cryptography service providers and libraries/classes:

- [SHA-256 \(.NET\)](#).
- [DSA \(.NET\)](#).
- [RSA \(.NET\)](#).
- [ECDSA \(.NET\)](#).
- [AES \(.NET\)](#).
- [Java Cryptography Class](#).

### The Case for Upgrading Toad

As a result of FIPS 140-2 compliance testing, Quest identified and addressed a vulnerability in Toad for Oracle v13.2, then notified customers as follows:

“Quest was made aware of an active exploit in the wild that allowed an attacker to decrypt credentials saved in the Toad product. The exploit, which took advantage of the way Toad encrypted passwords, could be used to decrypt and use credentials to compromise affected databases, FTP servers or SSH servers. Quest has since implemented strong encryption, patched this vulnerability and released a non-vulnerable version of the software.”

### CONCLUSION

The security controls described above are designed and applied to mitigate the risk of supply chain security in Toad products. Figure 2 illustrates their flow.

Toad products are valued and trusted, and they have unlocked millions of hours of productivity gains for database professionals. By upgrading Toad products, you ensure a continual flow of new features and adherence to software security standards. You also receive full technical support and keep your organization's security profile tight.

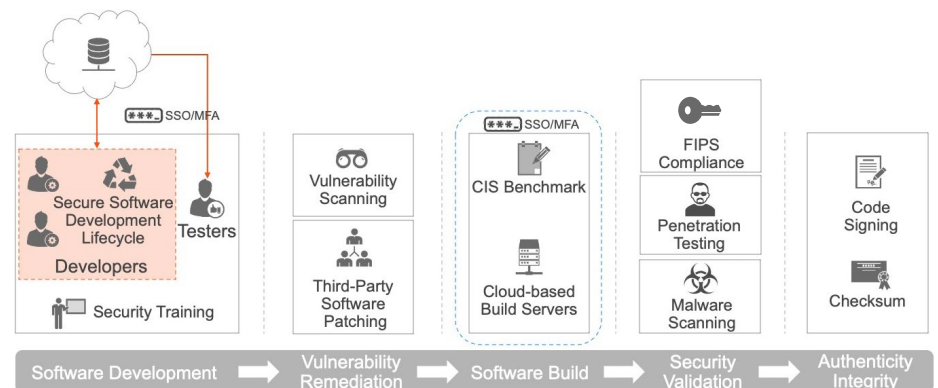


Figure 2: Flow of security controls in developing Toad products



## ABOUT THE AUTHORS

Julie Hyman is a senior product manager for the Database tools portfolio at Quest Software. She is a seasoned software product manager with 25 years of experience in creating and improving software products at startups and Fortune 500 firms. Julie has worked closely for years with DBAs, developers and analysts in all industries and many leading corporations to ensure Quest continues to provide world-class solutions.

John Pocknell is a senior market strategist at Quest Software and a member of the product marketing team. Based in the European headquarters in the U.K., John is responsible for synthesizing analyst data and customer interviews. He creates and evangelizes solutions-based stories and messaging related to major IT initiatives worldwide for Quest's portfolio of database products. He has been with Quest Software since 2000, working on products for database design, development and deployment, and he has spent over 10 years as product manager for Toad. John regularly evangelizes Quest's database solutions at conferences and user groups around the world and through blog posts and technical papers.

## ABOUT QUEST

Quest creates software solutions that make the benefits of new technology real in an increasingly complex IT landscape. From database and systems management, to Active Directory and Office 365 management, and cyber security resilience, Quest helps customers solve their next IT challenge now. Around the globe, more than 130,000 companies and 95% of the Fortune 500 count on Quest to deliver proactive management and monitoring for the next enterprise initiative, find the next solution for complex Microsoft challenges and stay ahead of the next threat. Quest Software. Where next meets now.

© 2020 Quest Software Inc. ALL RIGHTS RESERVED.

This guide contains proprietary information protected by copyright. The software described in this guide is furnished under a software license or nondisclosure agreement. This software may be used or copied only in accordance with the terms of the applicable agreement. No part of this guide may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying and recording for any purpose other than the purchaser's personal use without the written permission of Quest Software Inc.

The information in this document is provided in connection with Quest Software products. No license, express or implied, by estoppel or otherwise, to any intellectual property right is granted by this document or in connection with the sale of Quest Software products. EXCEPT AS SET FORTH IN THE TERMS AND CONDITIONS AS SPECIFIED IN THE LICENSE AGREEMENT FOR THIS PRODUCT, QUEST SOFTWARE ASSUMES NO LIABILITY WHATSOEVER AND DISCLAIMS ANY EXPRESS, IMPLIED OR STATUTORY WARRANTY RELATING TO ITS PRODUCTS INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT. IN NO EVENT SHALL QUEST SOFTWARE BE LIABLE FOR ANY DIRECT, INDIRECT, CONSEQUENTIAL, PUNITIVE, SPECIAL OR INCIDENTAL DAMAGES (INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF PROFITS, BUSINESS INTERRUPTION OR LOSS OF INFORMATION) ARISING OUT OF THE USE OR INABILITY TO USE THIS DOCUMENT, EVEN IF QUEST SOFTWARE HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. Quest Software makes no representations or warranties with respect to the accuracy or completeness of the contents of this document and reserves the right to make changes to specifications and product descriptions at any time without notice. Quest Software does not make any commitment to update the information contained in this document.

### Patents

Quest Software is proud of our advanced technology. Patents and pending patents may apply to this product. For the most current information about applicable patents for this product, please visit our website at [www.quest.com/legal](http://www.quest.com/legal)

### Trademarks

Quest, Toad and the Quest logo are trademarks and registered trademarks of Quest Software Inc. For a complete list of Quest marks, visit [www.quest.com/legal/trademark-information.aspx](http://www.quest.com/legal/trademark-information.aspx). All other trademarks are property of their respective owners.

If you have any questions regarding your potential use of this material, contact:

#### Quest Software Inc.

Attn: LEGAL Dept  
4 Polaris Way  
Aliso Viejo, CA 92656

Refer to our website ([www.quest.com](http://www.quest.com)) for regional and international office information.