
Data Governance in Software Testing

A Best Practice Primer

By Jon Collins, May 2008

In association with:



Introduction

In the research report “Data Governance in the Software Lifecycle” [1] we consider a number of areas of risk that can impact software development in general, and software testing in particular. The areas cover:

- Sourcing – how development teams are resourced, and with whom
- Geography – where developers reside, and the level of distribution within teams
- Environment – how the development, test and live systems environment is structured
- Data – what information is used in development and how it is managed

Each of these areas brings with it a set of specific risks which may need to be treated in some way, depending on the context. In this short guide we concern ourselves specifically with risks to do with the data used in software testing, and how they can be mitigated.

Much of what we lay out here may be considered to be basic common sense, but we make no apologies for this as our research indicates that some may not be fully aware of the issues and others need reminding. However much you feel you have this area under control, we encourage you to run through the guidelines and actions section on page 3 which has been designed to be used as a checklist.

Assessing the data related risk

There are several kinds of data that may be involved in the software testing process, including:

- Test scripts and guidelines
- Test results and metrics
- Configuration data
- Application data

When we look at this list, we may wonder what some of these have to do with security, so it is worth reviewing the potential areas of exposure.

The application being developed may give the business a competitive edge, for example, and test information taken as a whole may provide an indication of where that edge lies. In such cases, it may therefore be necessary to keep the very existence of the application under wraps, and therefore the entire test environment should be secured. Whether the application being developed is to support a new service or product line, or to enable entry into a new geographic market, the chances are that alerting competitors in advance of a launch via leaks or indiscretion within the

internal development organisation or a systems integrator can have a tangible negative impact on the business.

Beyond this, our research has shown [1] that the majority of organisations use data extracts from live systems in one form or another for testing purposes. While some of this data may be quite innocuous, other information might be extremely commercially sensitive and/or subject to legal or other regulatory constraints. Ensuring that such data is either appropriately sanitised or only accessible to those with the clearance to view it, is a fundamental part of security assurance within the software development and testing lifecycle.

Given the wide range of sensitivity that exists, a good first step is to classify each type of data according to its value and level of risk. There are many ways of doing this in practice, and classification systems in themselves can get quite complex. In order to facilitate effective communication across teams, however, we tend to recommend keeping things as simple as possible. Even a simple traffic light system may be adequate in many cases, such as the following:

- RED: Company proprietary or confidential, to be treated stringently
- AMBER: There are specific constraints on the data that need to be considered
- GREEN: There are no specific requirements to protect this data

Coming back to the nature of the risks themselves, it is then useful to firm up on the question of what exactly we are looking to protect data from. While the list of threats and exposures and the weightings given to each will vary by organisation (and even by project), here are some things to consider:

- Abuse of intellectual property
- Knowledge of system/process vulnerabilities falling into the wrong hands
- Engineering of vulnerabilities into applications for later exploitation
- Sabotage for personal or political reasons, and/or financial gain
- Unauthorised access to live systems or data
- Legal and compliance implications of live data use

This list is by no means exhaustive, but it is a useful starting point to identify where time and effort should be spent when considering the nature and level of exposure. Together with identifying the risk, of course, it is good practice to document the likelihood and potential impact of each.

As an approach, we would suggest using a format such as the following, though again, this is for illustration only as there may well be mechanisms already in place, either within IT or in more corporate level risk management systems that could be utilised:

Data Type	Classification	Instance and location	Risk and impact

Armed with this type of information, however it is captured, we can move on to how some of the risks can be treated or mitigated. In some situations, it might be appropriate to start with the question of whether the data in use is strictly necessary, but assuming it is, the principles of classification, risk assessment and access control will apply.

It is beyond the scope of this document to provide exhaustive advice on the appropriate action to be taken in each individual instance, but please see the guide on page 3 for a high level steer in the right direction.

References

[1] "Data Governance in the Software Lifecycle", Atherton, Collins and Vile, May 2008

Guidelines and Actions for Data Governance in Software Testing

Here we consider some general guidelines to mitigate data risks in software testing, derived from the report mentioned above [1], together with the specific actions that result.

Guideline	Rationale	Key Actions
Get up to speed on relevant legislation and compliance criteria.	Information about the test environment and live data used in the testing process may be subject to a number of standards and norms, which may in turn be legally binding. As a spin-off benefit, the compliance posture of the organisation may be enhanced.	<ul style="list-style-type: none"> Build a picture of what regulations and standards apply to the data in use in the test environment. Where snapshots of live application data are being used in testing, check the applicable regulations. Use the checklist of test information (below) to cross-reference whether the list of applicable regulations and standards is complete.
Build a picture of all data at use in the test environment, both current and historical	When identifying test-related data in play, it is too easy to look only at material currently in use. However, some historical information may be still accessible, either in an archive or potentially just sitting in the file system (or indeed a filing cabinet).	<ul style="list-style-type: none"> Create an inventory of the different types of test-related information in use, and assess it using the traffic light system described above. Decide on the risks, likelihoods and impacts for each element of test information, taking into account the regulations and standards.
Employ a mix of centralised and local policies for data handling.	Where only central policy is involved, it can be quite separate from the 'coal face' and hence is sometimes seen as irrelevant or incomplete. Local policies should complement central policies to ensure completeness of coverage.	<ul style="list-style-type: none"> Review central policies to ensure they cover the needs of the testing environment Create a local testing policy document to cover the specific requirements of local testing. Assign responsibility for local policy implementation and review. Gain buy-in from both testers and management to maximise the effectiveness of policy implementation.
Review test processes for risk mitigation and consistency with standards/regulations.	Test processes should take into account not just testing completeness and consistency, but also the security of test information. A weak process puts data at risk.	<ul style="list-style-type: none"> Collate a complete set of test processes for the testing at hand. Verify that testing processes are being followed consistently (if not, that's a security risk in itself). Recommend and apply improvements to test processes where they prove to be inadequate.
Define appropriate roles to oversee the testing process from a security perspective.	Security implementation in general, and in the test environment in particular, works best when it is being treated as a priority by the right level of seniority within the organisation. Business involvement is particularly effective.	<ul style="list-style-type: none"> Review existing security roles and responsibilities and identify where the role for test data management would best fit. Check compliance and standards for specific requirements to incorporate responsibilities into other roles in the organisation as necessary.
Deploy appropriate processes and tools for mitigation of test data risks.	Certain types of data, for example customer data, may be at risk (or even illegal) if used in their raw state in the test environment.	<ul style="list-style-type: none"> Identify if and where live application data is being used that may pose a risk when used in the test environment Review test data management processes (where these exist) to minimise the use of such data and ensure it is correctly managed. Consider tools to assist with the sanitisation of such data to further reduce live data risks.
Destroy unnecessary or obsolete data.	Data may be retained unnecessarily, creating a management overhead and an ongoing risk.	<ul style="list-style-type: none"> Take a careful look at the data being stored. Check it is still applicable, and indeed usable. If not, then delete or otherwise destroy it. If in doubt, quarantine it – i.e. collate it and store it in a secure place.

About Freeform Dynamics



Freeform Dynamics is a research and analysis firm. We track and report on the business impact of developments in the IT and communications sectors.

As part of this, we use an innovative research methodology to gather feedback directly from those involved in IT strategy, planning, procurement and implementation. Our output is therefore grounded in real-world practicality for use by mainstream business and IT professionals.

For further information or to subscribe to the Freeform Dynamics free research service, please visit www.freeformdynamics.com or contact us via info@freeformdynamics.com.

Terms of Use

This document is Copyright 2008 Freeform Dynamics Ltd. It may be freely duplicated and distributed in its entirety on an individual one to one basis, either electronically or in hard copy form. It may not, however, be disassembled or modified in any way as part of the duplication process. Hosting of the document for download on the Web and/or mass distribution of the document by any means is prohibited unless express permission is obtained from Freeform Dynamics Ltd.

This document is provided for your general information and use only. Neither Freeform Dynamics Ltd nor any third parties provide any warranty or guarantee as to the suitability of the information provided within it for any particular purpose.