Spreadsheet Assurance

Tom Olzak
May 2006

Electronic spreadsheets are used throughout the business sector—and use is growing steadily. Many organizations supplement financial applications with spreadsheet calculations. Others simply use spreadsheets instead of specialized software. A useful tool—and one of the main drivers that catapulted personal computers into the forefront of business data processing—the electronic spreadsheet can introduce significant risk.

In this paper I look at the challenges faced by organizations that manage by spreadsheet. I'll also examine ways to secure and manage spreadsheets while in production.

The Challenges

According to Dr. Raymond Panko, most spreadsheets contain errors (2005). Statistically, spreadsheets with large numbers of calculations have an error rate approaching that of application source code—about 5%. The actual error rate in your spreadsheets depends primarily on the level of expertise of the individuals developing them and the complexity of the spreadsheets.

To date, auditors have not looked too closely at spreadsheets that support financial management. However, it's just a matter of time until they begin to investigate ***all*** tools used to manage financial information.

SOX isn't the only regulation affecting spreadsheet users. The Health Insurance Portability and Accountability Act (HIPAA) regulates the confidentiality and availability of Protected Health Information (PHI). Since businesses don't centrally manage user spreadsheets, healthcare companies might find that protecting PHI in spreadsheets is a massive task.

Will your spreadsheet results pass an audit? Do you have safeguards in place to protect the confidentiality, integrity (accuracy), and availability of information contained in your users' spreadsheets? Before proceeding on to the rest of this paper, try to answer the following questions about your spreadsheets that process or contain critical or sensitive information (Freeman, 1996).

1. What controls prevent errors from appearing in the output?
2. Does each spreadsheet rely on the knowledge and skills of just one individual? If so, what happens if that person is no longer available?
3. Is there an audit trail to show each change to each spreadsheet?
4. Does documentation exist that explains the design of the spreadsheet and instructions on how to use it?

5. Was the spreadsheet tested? Has the test data been retained to ensure that the data are still processed properly when the spreadsheet is altered?
6. What ensures that data entered into the spreadsheet do not contain errors or inconsistencies when compared with their sources?
7. How does a user know she's using the right version of the spreadsheet?

If you answer these questions honestly, you should have a good idea about the level of risk your organization faces because of spreadsheet use. Thinking through the answers also provides insight into the kinds of safeguards you should consider. Let's look at some controls for protecting your data and your business.

Implement a Spreadsheet Development Lifecycle Process

Since developing complex spreadsheets is similar to developing applications, you should consider following a development lifecycle process to ensure data integrity. Although you might not buy in to all System Development Lifecycle (SDLC) steps, you should consider the following:

1. Create an overall description of the spreadsheet to be developed. Include the business challenges addressed, who will use the spreadsheet, the kinds of data it will contain, who will own the spreadsheet, etc. As with production applications and data, having this information allows management to determine the level of security required, how tightly to control access, and who should have rights to change the calculations and format of the spreadsheet.
2. Clearly define the functional and technical requirements to be met. When defining requirements, consider including the following controls:
    - Create separate areas for calculation cells and data entry cells
    - Protect calculation areas from modification
    - Audit all changes
    - Ensure effective version control
3. Develop the spreadsheet to meet all requirements. Defined within the context of spreadsheet development, a developer can be a business user, manager, or an actual application developer. Since the skill level of the developer has a direct impact on error rate, a business should carefully consider who is given responsibility for creating each business critical or regulated spreadsheet.
4. Conduct a review of spreadsheet layout and calculation accuracy. This review should be accomplished by someone other than the developer. Once testing is complete, the calculation cells should be locked, a version number assigned, and the spreadsheet forwarded for user acceptance testing. At no time should the spreadsheet return to the developer after testing unless the version number changes.
5. When testing is complete, get signoff from management and user stakeholders.
6. Vault the approved version. In other words, lock the version that you're moving to production by placing it into some kind of document management system. The extent to which you apply version control depends on the criticality of spreadsheet contents. Like anything else, you should strike a balance between assurance, cost, and operational efficiency.

OS and Application Security

In a Windows/Excel environment, the first layer of protection is NTFS. (We're assuming that Share permission is set to allow change access for all authenticated users.) Organize your spreadsheets into a folder structure designed to allow the use of groups for folder access. For example, spreadsheets used by both Finance and Accounting might be in a single folder. A local group is given read/write permissions to the folder. The Finance and Accounting Active Directory global groups are placed into the local group. All other non-administrator access is removed. At this point, only members of the Finance and Accounting groups have access to the files.

The next step is to determine if all members of the Finance and Accounting global groups require access to all the spreadsheets in the folder. If not, there are two solutions. One solution involves creating additional folders with additional local groups to refine access. But this can create a security structure that's hard to manage. Another approach is to use Excel security to password protect each workbook with special access restrictions.

Figure 1 shows the Excel form used to assign open and modify permissions to a spreadsheet. This window is accessible by selecting Options from the Tools menu, and then clicking on the Security tab. Entering a password into the *Password to open* field causes the Excel file to be encrypted when it's saved to the spreadsheet folder. This is a good solution when you have a relatively small number of file access exceptions in your spreadsheet folder. It's also the strongest Excel protection available through the application. Cracking an eight character password to open a workbook file, for example, requires a work factor sufficiently onerous to deter most attackers.

A final layer of Excel protection consists of workbook structure passwords, worksheet passwords, and VBA passwords intended to provide weak protection against changes by authorized users. However, Microsoft warns against relying on these controls to deter determined attackers—both internal and external.

"Excel features related to hiding data or locking data with passwords are not intended to secure or protect confidential information in Excel. These features are merely meant to obscure data or formulas that might confuse some users or to prevent others from viewing or making changes to that data" (Microsoft, 2006).

The Excel features referred to in the above quote are easily cracked with free software readily available on the Internet.

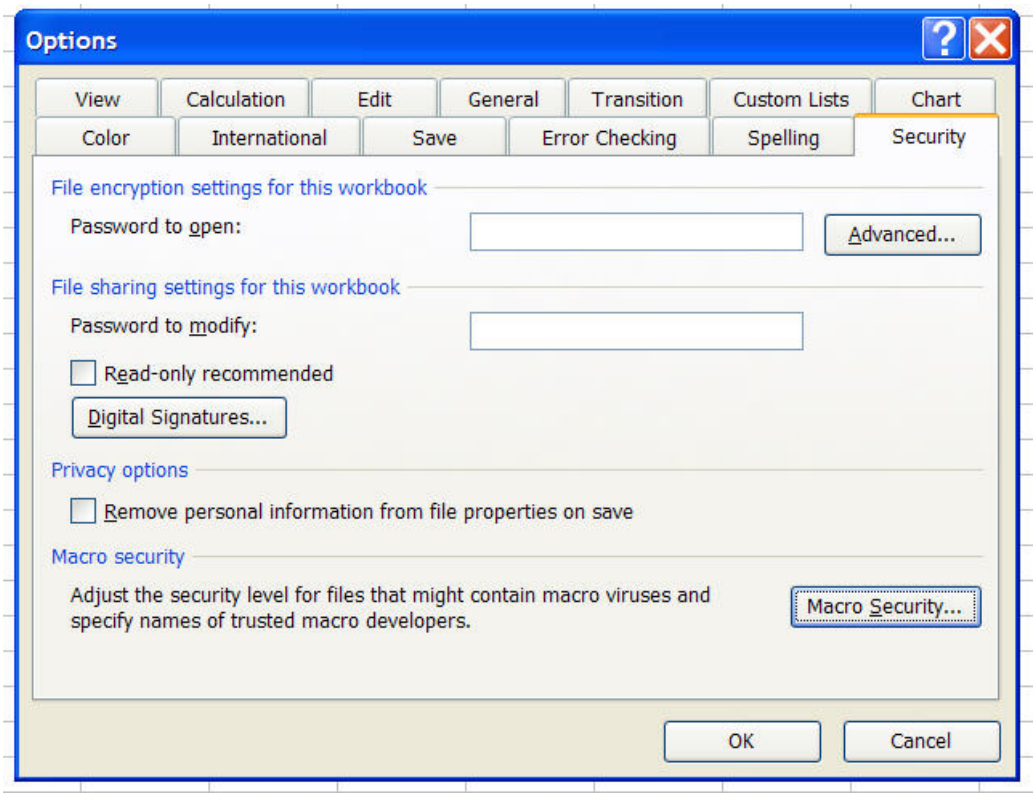

Figure 1: Excel Options Security Tab

Application and OS controls are fine for file access management, but they don't provide the granularity required to create a detailed audit trail or to adequately prevent worksheet changes by individuals with authorized access to a workbook file. This is a gap that third party software providers are trying to fill.

Third Party Tools

If you have the resources, you might consider replacing OS and application controls with third party tools designed to manage your critical spreadsheets. One such product is Remediation Services for Microsoft Excel (RSME) from Agilent Technologies. Management of your spreadsheets through RSME provides the following features:

- Comprehensive file, system, and cell-by-cell audit trails
- Electronic signature support
- Version control and revision storage for spreadsheets
- Differencing tool for compare changes between spreadsheets
- Access control based on users, groups, roles, and privileges
- Spreadsheet security and data integrity
- Archival and backup support
- Applies effective change controls and security

Conclusion

No matter how you plan to provide spreadsheet assurance, be sure to start taking steps to protect your sensitive information. Educate management about the perils of using spreadsheets for processing and storage of business critical information, and implement SDLC methodologies for managing complex spreadsheet creation and modification.

Additional Information

European Spreadsheet Risks Interest Group—http://www.eusprig.org/stories.htm

Works Cited

Freeman, D. (1996). How to make spreadsheets error-proof. *Journal of Accountancy, 18*(5), 75.

Microsoft (2006). *Overview of security and protection in Excel.* Retrieved May 4, 2006 from http://office.microsoft.com/en-us/assistance/HP052388541033.aspx

Panko, R. (2005, January). *What we know about spreadsheet errors.* Retrieved May 1, 2006 from http://panko.cba.hawaii.edu/ssr/Mypapers/whatknow.htm