OpenOffice v3.x Security Design Weaknesses

Eric Filiol filiol@esiea.fr
Jean-Paul Fizaine fizaine@esiea-ouest.fr

Ecole Suypérieure en Informatique, Electronique et Automatique
(ESIEA - Laval)
Operational virology and cryptology Lab.

esiea

INTRODUCTION

- For years Microsoft Office has been THE reference suite
  - For document production.
  - For document exchange.
- Very soon infested by macro-viruses.
  - Concept virus (1995).
- Still a real threat.
  - E. g. China vs German chancery (2007).
- Need for an alternative?

INTRODUCTION

- Recent evolution
  - Use free Office suite
  - The best candidate: OpenOffice.
- Very popular:
  - Seemingly no cost.
  - Wrong feeling of security
    « *It is free and open therefore it is (or must be) secure!* »
  - Fully compatible with Microsoft Office
  - … more than Microsoft with itself.
- Worldwide use in civilian and governmental (incl. military) spheres. Official document format for:
  - French Gendarmerie, French Ministry of Economy and Finance
  - And many others in Europe…

esiea

INTRODUCTION

- The "natural" confidence in Open Software makes security analysis most of the times useless.
- Question: it is possible to have both security and openness at the same time?
- What the exact level of security with respect to malware when considering OO.
- BadBunny macro worm (2008).

esiea

INTRODUCTION

- In 2006 and 2007 security analysis showed that OO 2.x was absolutely not secure.
  - All data given to OO developers
- End of 2008, release of OO3
  - Presented as a significant evolution!
  - What about security two years after?

esiea

INTRODUCTION

- Our talk deals with an in-depth analysis of OO3 with respect to malware attacks
  - How to exploit the confidence in cryptographic primitives?
  - How to design powerful attacks?
- We do not consider implementation vulnerabilities!
- We consider conceptual design flaws only!
- Wlog we consider OOwriter only!

esiea

INTRODUCTION

- To prevent stupid comments:
  - We are not hidden Microsoft moles!
  - There are problems for M$ too.
  - But unfortunately less than for OO since it has less powerful primitives inside.
- We just want to make decision-makers to be aware of the existing risks
  - … and make them responsible, if such a thing is possible!
  - Reducing costs is most of the times not compatible with security.

esiea

AGENDA

- Introduction.
- History of OO 2.x security.
- ODF Format and Security Primitives.
- Viral Attacks through OO3 documents
  - Unencrypted documents
  - Encrypted documents
  - Digitally signed documents
- Conclusion: Enhancing OO Security.

esiea

Demos

- A lot of demos to come.
- Complete code and techniques available in the white paper!
- Fully and easily implementable by malware in an automatic way.

esiea

History of OO 2.x security

Black Hat Briefings

OO2 Security History

- **First in-depth security analysis**
  - *De Drézigué et al. (2006) Journal in Computer virology*
  - *Filiol & Fizaine (2007) Virus Bulletin Journal.*
  - *Lagadec (2007) Journal in Computer Virology*
- **A lot of « hot » reactions.**
  - **Many stupid, ideologic comments but who did really read the papers?**

esiea

OO2 Security History (2)

- **A lot of contacts with the OO developers (German part)**
  - *All proof-of-concepts communicated to them during the OO International Conference in Lyon, France (2006).*
  - *We suggested to design the Trusted OpenOffice suite:*
    - *Parts or sensitive functions of the suite could be enabled/disabled by the system admistrator according to the security policy in place.*
- *To answer the permanent stupid comments, we published technical data (Virus Bulletin).*

OO2 Security History (3)

- **OpenOffice malware appear**
  - *Proof-of-concept (Filiol & Fizaine, 2006 & 2007).*
  - *BadBunny (2007).*
  - *What about the next ones?*
- **Unfortunately, results are not taken into account!**
  - No real security concern.
  - OO embed cryptography!
  - *The OO suite « spreads » more and more.*

SOPHOS

21 mai 2007

Ver SB/BadBunny-A

SB/BadBunny-A est un ver multi-plates-formes écrit en de nombreux langages scripts et distribué comme un **document OpenOffice.org** contenant une macro StarBasic.

SB/BadBunny-A se propage en injectant des fichiers script malveillants qui affectent le comportement de programmes IRC, mIRC et X-Chat populaires et provoquant l'envoi de SB/BadBunny-A à d'autres utilisateurs.

Ces fichiers scripts malveillants sont nommés badbunny.py (pour XChat) et script.ini (pour mIRC, écrasant le fichier mIRC existant) et sont aussi détectés sous le nom de SB/BadBunny-A.

SB/BadBunny-A injecte différents composants supplémentaires sur la plate-forme sur laquelle il s'exécute :

- Sur **Windows**, il injecte un fichier nommé badbunny.js qui est un infecteur de fichier JavaScript aussi détecté sous le nom de SB/BadBunny-A.
- Sur **Linux**, il injecte un fichier nommé badbunny.pl qui est un infecteur de fichier Perl aussi détecté sous le nom de SB/BadBunny-A.
- Sur **MacOS**, il injecte un ou deux fichiers nommés badbunny.rb et badbunnya.rb qui sont des infecteurs de fichiers Ruby aussi détectés sous le nom de SB/BadBunny-A.

esiea

OO2 Security History (5)

- Which attacks were possible?
  - Macro management modification:
    - Change or pervert the macro security level
    - Possibility to insert malicious macros in OO libraries
  - Modification of the application menus (problem of application integrity management). Interesting to use with k-ary malware.
  - Modify integrity of plain document (insert macro)
- Weak management of cryptography. Possibility to transparently remove:
  - Encryption.
  - Digital signature.

esiea

OO2 Security History (6)

- We designed proof-of-concepts for technical validation.
- We will not present the viral algorithmics:
  - Not specific to OO but to macro viruses
  - With OO3, nothing has really changed with respect to the malware technologies
  - Please refer to the bibliography.
- We are going to explain how to exploit user's confidence in cryptography (encryption, signature) to design powerful malware attacks.

esiea

OO3 Release

- December 2008: release of OO3
- Presented as a major evolution of the suite
  - Compatibility with Vista!
  - A few bugs fixed
  - Easy-to-useness increased
  - …
- But what about security?
- Are cryptographic (encryption, signature) a real protection against OO malware.
- In fact most of the attacks still remain effective!

ODF Format and Security Primitives.

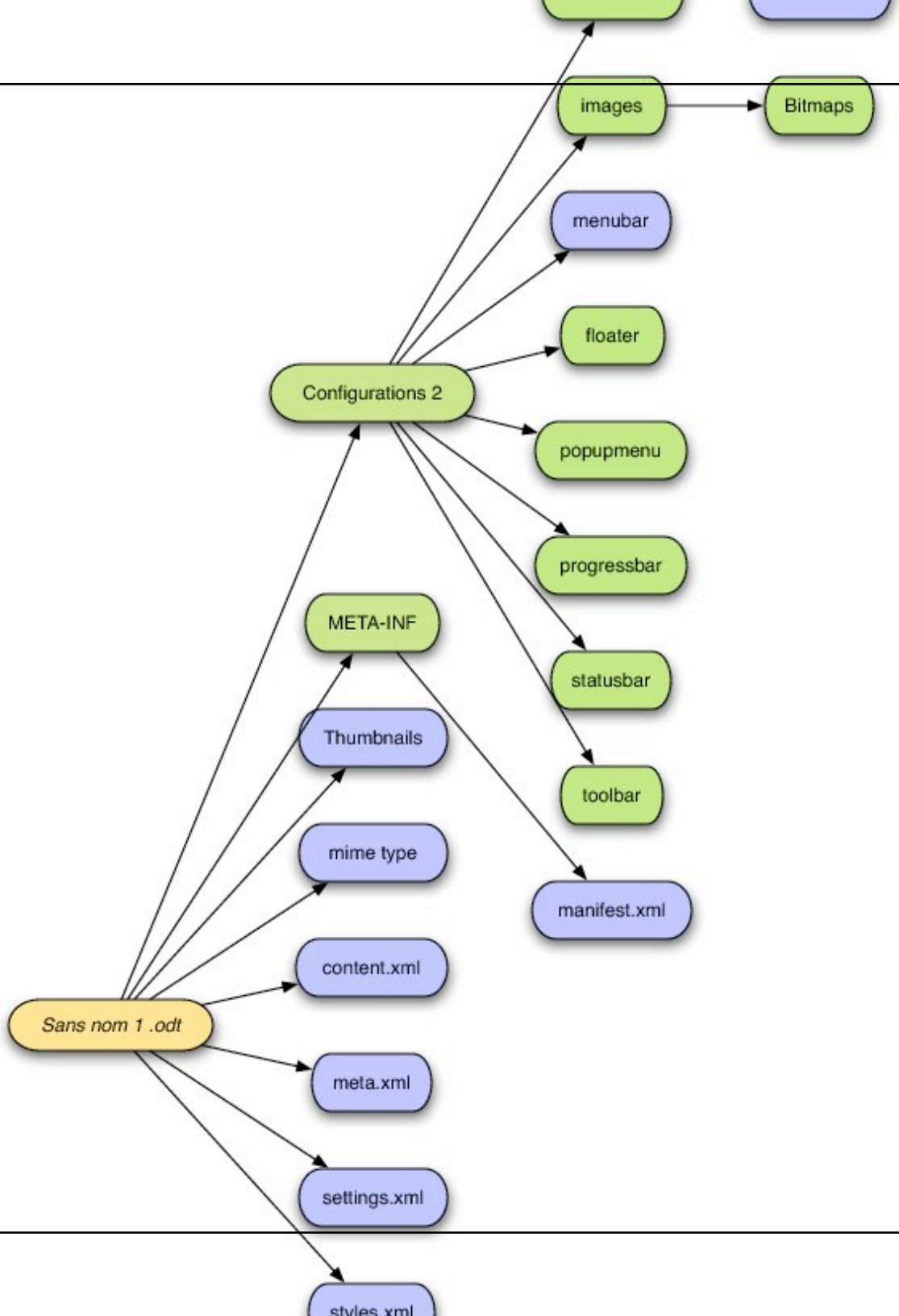
ODF - Formal approach

Black Hat Briefings

esiea

- **Any (**
- **With**
  editc
  such
  - *De*
- **Two**
  - ***Co***
  - ***Ma***
    (f
  - *De*

, text
fy any
ure



esiea

OO3 Macro Location

- **Where are located macros in OO3 documents?**
  - *Located in a specific directory (one per language).*
  - *Contains the files*
    - *Script-lb.xml (generic information with respect to macros)*
    - *Script-lc.xml (additional information + security flags)*

    Library:readonly="false"

    Library:passwordprotected="false"
  - **The macro code itself!**
  - Demo 3

esiea

OO3 Cryptographic Features Formalization

- OO3 security is based on
  - Password-based encryption.
  - Digital signature.
- There are (too) many ways to apply them.
- Need for a formal approach for an exhaustive description.
- Graph-based description
  - Digital signature
  - Digital Signature with encryption.

OO3 Encryption

- **Blowfish in CFB mode.**
  - *Use of IV for key differentiation!*
  - *In this respect far better than M$ Office (Filiol, 2009).*
- **Key derivation algorithm***: PBKDF2*
- **SHA-1 for integrity.**
- **The manifest.xml file is itself not encrypted!**
  - *Major weakness that can be exploited by malware***!**
- *Demo 4*

esiea

OO3 Encryption (2)

- **Macro and macro-related files are themselves encrypted**
  - Demo 5
- As we will see, it is only an apparent protection in most critical cases.

esiea

OO3 Signature

- **Let us recall that signature is THE cryptographic primitive dedicated to give confidence about**
  - **Document integrity**
  - **Document origin (who is the sender)**
- **There are two ways of applying signature**
  - ***File → Digital Signature…* menu**
  - ***Tools → Macros → Digital Signature*… menu**
- **Based on X509 certificates**
  - Demo 6 (signature of document without macro)
  - Creation of a *documentsignatures.xml* file
  - Both the *manifest.xml* and *documentsignatures.xml* *files are not signed!*

esiea

OO3 Signature and Encryption

- **The overall structure remains the same.**
  - **Refer to the white paper.**
- **The** *documentsignatures.xml* is not encrypted!
  - Another critical weakness!

- Let us now consider documents with macros.
  - Two different cases to consider!
  - But in both cases the critical files are not signed!

File → Digital Signature Case

- **A *documentsignature.xml* file is created**
- **The whole document is signed (including macros) !**
  - Significant evolution compared to OO2.
  - Older attacks now fails!
  - But new ones are possible (see further)!
- Demo 7

esiea

Tools → Macros → Digital Signature Case

- **A *macrosignatures.xml* file is created**
- **Only the macro tree is signed (including the macros)**
  - Possible to modify the rest of the document while the user relies on partial signature!
  - Other attacks are possible with respect to macros (see further).
- Demo 8

esiea

Summary

- **There is still critical weaknesses with respect to signature and encryption implemenation/ management.**
  - A few older attacks from 2006/2007 are no longer directly valid.
  - New ones are possible.
- **The existence of two different methods for signature is non sensical and is bound to fool the user and ease malware attacks.**
- **Cryptographic primitives provides a false sense of security to the user!**
- **Let us now explain why.**

esiea

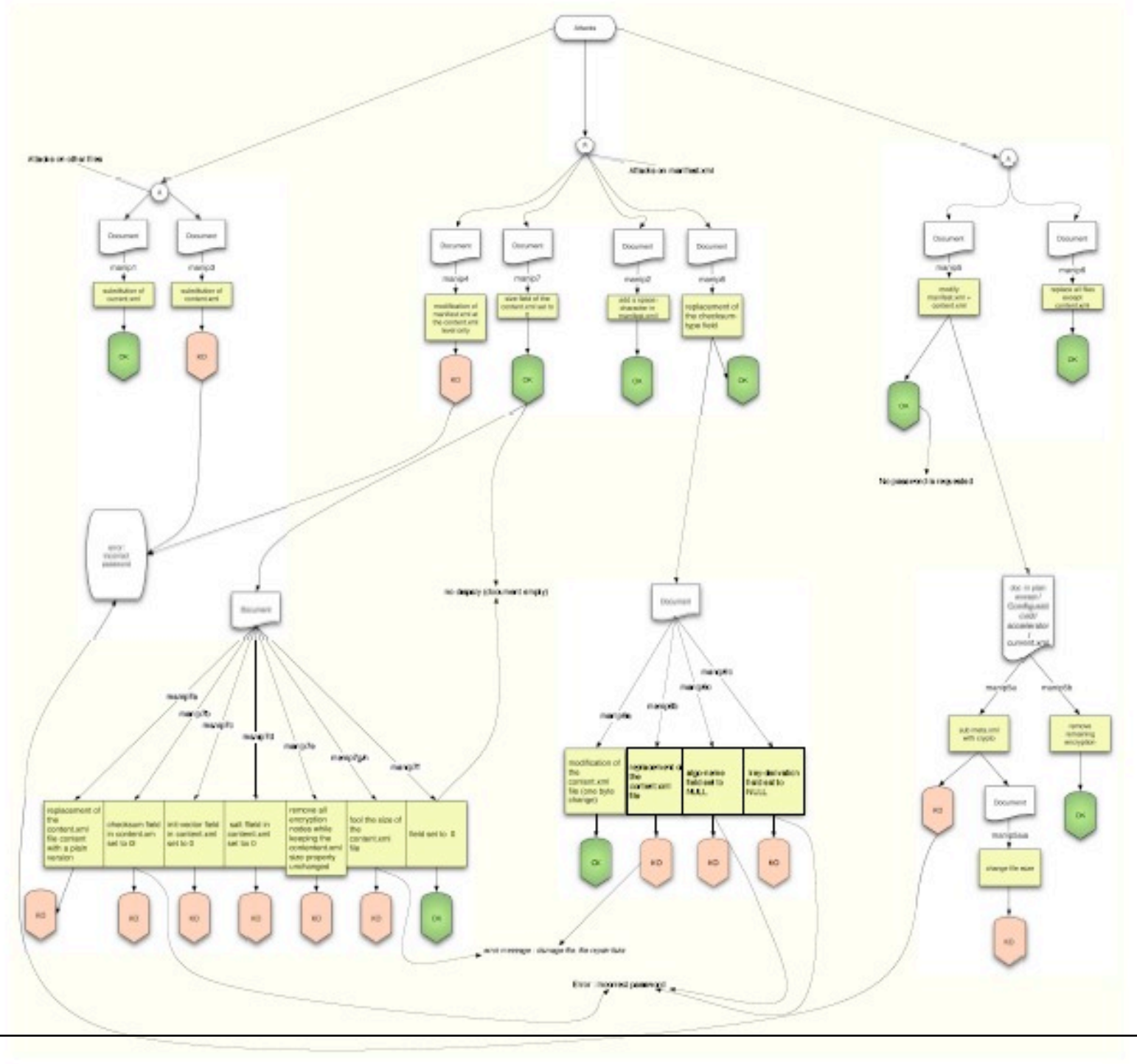
Viral Attacks through OO3 documents

Black Hat Briefings

esiea

Through Unencrypted Documents

- **No significant changes compared to OO2.**
- **Simple archive manipulations allow to perform a lot of attacks.**
  - Modify the content.xml file (*demo A1*).
  - Add files. Useful for document theft.
  - Add macro.
  - Substitute macros (*demo A2*)
- **No integrity management at all.**
- **OO3 plain documents are very powerful malware vectors.**



esiea

Through Signed Documents

- **Significant changes compared to OO2.**
- **It is no longer possible to**
  - Add a macro to a signed document
  - Replace a macro with another (malicious) macro.
- **BUT OO3 signature provides the illusion of security only!**
- **Since there is no PKI yet to securely manage signature:**
  - Man-in-the-middle attacks are very easy to revert trust against the user
  - **Demo A4**

esiea

Through Signed Documents (2)

- Alice signs her document.
- Charlie the attacker forges a Alice's fake x509 certificate
  - Very easy to recover the necessary information.
  - Just read the *meta.xml* file (possibly of in a previous document).
- Charlie generates a Alice's fake pair of keys and signs the document in Alice's name (impersonation attack) after adding malicious macros.
- Bob the receiver checks the signature and is fooled.
- A close look at certificates (Demo A5).

Conclusion

Enhancing OO3 Security

esiea

Protection measures

- **Postpone use of OO3 for critical use!**
- **Use external signature modules with PKI.**
  - **French project Linagora (Open cryptographic component EAL3+)**
  - **http://wiki.services.openoffice.org/wiki/Improving_the_digital_signature_Feature**
- **Apply security policy rules**
  - **Control of origin**
  - **Control of contents**
  - **…**

esiea

Change the Design

- **Files manifest.xml and meta.xml should be encrypted to prevent information extraction.**
- **Semantic verification of the archive should be implemented**
  - **At the present time only the XML specification syntax is checked.**
  - **Implement λ-calculus-based techniques!**
- **Design the Trusted OpenOffice suite**
  - **Enable/disable functions/languages through an administrator password.**
- **… or use LaTeX!**

Thanks for your attention

Questions ?