DATA LOCKDOWN

Data Loss Prevention
Through Steganography
About the Speaker

Anne Shultz

- Graduate Student at IIT
  - Information Technology & Management Program
- Previous experience working in IT Security for a large company (auto-manufacturing plant)
  - Learned to appreciate the challenges of securing unstructured data!
Project Goals

1. Create a system allowing users to **tag** sensitive unstructured data

2. Develop a method to **check** whether data has been tagged

3. **Secure** tagged data by preventing it from leaving a network
What is Unstructured Data?

- Data which is not stored in a database
- Electronic documents where the contents can take any shape
Why is Unstructured Data Security a Problem?

- Over 85% of business information is made up of unstructured data
  - Estimated by Merrill Lynch (Atre and Blumberg, 2003)

- 89% of respondents admit controlling access to unstructured data is difficult for their company
  - Reported by Ponemon Institute and Varonis (StorageNewsletter.com)
Existing Solutions...

- Data Loss Prevention (DLP) Systems
  - Identify sensitive data
  - Send alerts when this data passes outside the company’s network

- However...
  - Methods used for identifying sensitive data are flawed
DLP Devices Currently...

- Scan for particular character strings
  - False positives!
- Can upload sensitive documents to ensure matching documents can be identified
  - Excessive management!
Mainly...

- A user, creating a document, cannot easily label whether their document is sensitive!
Step 1: Create a system allowing users to tag sensitive unstructured data

Method:
- Use **steganography** to hide a “tag” file inside MS Office 2007 Word, PowerPoint, & Excel Files

How?
- **Windows Powershell**
  - Command-line shell and scripting language (by Microsoft)
Step 2: Develop a method to check whether data has been tagged

**Method:**
- Use **steganalysis** to extract a “tag” file from MS Office 2007 Word, PowerPoint, & Excel Files

**How?**
- Windows Powershell
  - Command-line shell and scripting language (by Microsoft)
Step 3  Secure tagged data by preventing it from leaving a network

Method:
- Modify an **SMTP Proxy Server** to check for “tag” files in outgoing email attachments (.docx, .pptx, and .xlsx)

How?
- **VBScript**
  - Active Scripting language (by Microsoft)
- **Windows Powershell**
  - Command-line shell and scripting language (by Microsoft)
Why Steganography?

- Can be applied to an individual document
- Can be applied locally by the user
  - (if tools are provided)
- Can accommodate a variety of file types
  - Project scope limited to .docx, .pptx, and .xlsx
Steganography for .docx, .pptx, & .xlsx

Microsoft Office 2007 file = Compressed file package
Inside a MS Office 2007 File...

This file contains:
1. ~~~~~~
2. ~~~~~~
3. ~~~~~~

.docx, .pptx, or .xlsx
To add a tag...

1. Just unzip the MS Office 2007 file
2. Add the tag file
3. Add the name of the tag file to the [Content_Types].xml file
4. Re-zip the MS Office 2007 file!

Right?

Wrong!
Why?

- MS Office 2007 files are not compressed in the same way as “zip” files
- If you try to “zip” the file back up,
  - The file will be corrupted!

**SOLUTION:**
- You must change the contents of the MS Office 2007 file *without unzipping it!*
How?

- **DotNetZip Library**
  - free class library and toolset for manipulating zip files or folders
  - From CodePlex (Open Source Project Community)

- Allows you to add files to a MS Office 2007 compressed file without unzipping it!
Create a system allowing users to tag sensitive unstructured data

Step 1
Step 1 Includes...

- **Powershell Scripts**
  - ScanDirectory.ps1
  - CheckAddErrors.ps1
  - Add.ps1

- **DotNetZip Library**

- **MS Office 2007 files**

- **Tag Files**
  - Tag1.txt ("PUBLIC" Tag)
  - Tag2.txt ("INTERNAL USE ONLY" Tag)
  - Tag3.txt ("COMPANY CONFIDENTIAL" Tag)
Tag Files:

Tag1.txt - Notepad
PUBLIC

Tag2.txt - Notepad
INTERNAL USE ONLY

Tag3.txt - Notepad
COMPANY CONFIDENTIAL
ScanDirectory.ps1

- **Scans** last modified dates for .xlsx, docx, and .pptx files **every 5 seconds**
ScanDirectory.ps1

- If a last modified date has changed, the file is sent to CheckAddErrors.ps1
Why CheckAddErrors.ps1?

- The Add.ps1 script cannot access the MS Office file if it is still open.
- The script will throw an error.
CheckAddErrors.ps1

1. Attempts to access the file and catches the error if the file is still open

2. If an error is caught, user is prompted →

3. If no error is caught, (or if “OK” is selected), the file is sent to Add.ps1
Add.ps1

1. Prompts the user:
Add.ps1

2. Finds & Modifies [Content_Types].xml file,

3. Adds the correct tag,

[Content_Types].xml

This file contains....

1. ~~~~~~
2. ~~~~~~
3. ~~~~~~
4. Tag3.txt
Add.ps1

4. Displays a Pop-up message

Thank You!

SecretStuff.docx has been tagged.

OK
Develop a method to check whether data has been tagged

Step 2
Step 2 Includes...

- Powershell Scripts
  - Detect.ps1

- DotNetZip Library

- Added Registry Keys
  - To enable execution of Detect.ps1 from the right-click menu

- Tagged MS Office 2007 files
So,

- When a user **right clicks** on a `.docx`, `.pptx`, or `.xlsx` file,
- And selects **“Check Sensitivity Level,”**
- **Detect.ps1** is executed...
Detect.ps1

- Unzips the file to a temporary folder
- Checks for a tag file

This file contains:
1. ~~~~~~
2. ~~~~~~
3. ~~~~~~
4. Tag3.txt
Detect.ps1

- If a tag exists, tag contents is printed

- If no tag exists, a pop-up reads as follows
Issue!

- File names & paths with spaces cause Detect.ps1 to fail
  - The registry command uses "%1" to pass the file name & path to Detect.ps1
  - If file name or path has a space, "%1" will only expand until the space

Example

C:\Documents and Settings\User\Desktop\MyDoc.docx

Is passed as:

C:\Documents
Issue!

Temporary Solution:

To Check Sensitivity Level:

This document must be saved with a filename & filepath that does not include spaces.
Secure tagged data by preventing it from leaving a network

**Step 3**
Step 3 Includes...

- User Desktop
  - With ability to apply and check tags
  - Email Client (Mozilla Thunderbird)

- Manager Desktop
  - Email Client (Mozilla Thunderbird)

- SMTP Server (hMailServer)
  - VBScript
    - EventHandlers.vbs
  - Powershell Script
    - ScanFile.ps1
User sends an email with a tagged attachment
Example email
(sent from BadUser@datalockdown.com)

Hi Anne,

Please find my totally secret, sensitive data attached! (Don't tell anyone I sent this!!)

-Bad User
SMTP Server checks for tagged attachment
How? EventHandlers.vbs

- Provided by hMailServer
  - Script for setting event handlers
  - “On Message Accept” event has been modified for this project

1. **On Message Accept**, checks for attachments with .docx, .pptx, and .xlsx
2. If attachment exists, sends file to ScanFile.ps1
ScanFile.ps1

1. **Unzips** the file to a temporary folder
2. **Checks** for a tag file

This file contains:
1. ~~~~~~
2. ~~~~~~
3. ~~~~~~
4. Tag3.txt

---

Confidential Tag

Tag3.txt

[Content_Types].xml

Document Parts
3. If tag3.txt or tag2.txt exists,

4. Return 1 to EventHandlers.vbs

5. Send Alert to Manager & Sender
Alert is sent to the User and the Manager
Example Manager Alert
(to Manager@datalockdown.com)

User at email address BadUser@datalockdown.com has attempted to send SecretStuff.docx. This file has been identified as COMPANY CONFIDENTIAL and has been blocked from leaving the company network.
Example Sender Alert
(to BadUser@datalockdown.com)

SECURITY ALERT: SecretStuff.docx has been identified as COMPANY CONFIDENTIAL and must not be sent outside the company. This file has been removed from your outgoing email. Management has also been notified of your attempt to send this file.
Meanwhile... EventHandlers.vbs

- Receives 1 from ScanFile.ps1
  - Deletes attachment from email
  - Sends email as normal

- If 1 is not received,
  - Sends email as normal
Issues

- This product is **not ready** for implementation in an organizational setting

- Many “bugs” still exist
  - ScanDirectory.ps1 is a memory hog
  - Powershell scripts cannot access encrypted MS Office files
  - When you open and resave an MS Office file, it must be re-tagged
  - To name a few…
Perspectives

- Nonetheless, this project serves as a proof of concept
  - *This IS possible!!!*

- Opens a realm of possibilities for using steganography and network security to track and secure sensitive data
Future Areas of Research

- Resolving remaining “bugs”
- Adapting the scripts to include additional file types (such as .pdf, .vsd, .jpg, etc.)
- Adjusting the scripts for easy modification of tag options
- Scalability for networks with large numbers of users
Questions?