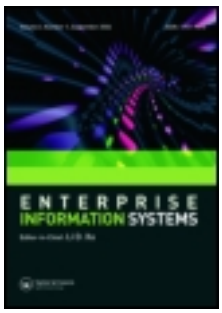

Enterprise Information Systems
Publication details, including instructions for authors and subscription information:
http://www.tandfonline.com/loi/teis20

Controlled information destruction: the final frontier in preserving information security for every organisation

Daniel-Ioan Curiac [a] & Mihai Pachia [a]

[a] 'Politehnica' University of Timisoara , Automation and Applied Informatics , Bd. V. Parvan nr. 2, Timisoara, 300223 , Romania

Published online: 16 May 2013.

To cite this article: Enterprise Information Systems (2013): Controlled information destruction: the final frontier in preserving information security for every organisation, Enterprise Information Systems, DOI: 10.1080/17517575.2013.792397

To link to this article: http://dx.doi.org/10.1080/17517575.2013.792397

Enterprise Information Systems, 2013
http://dx.doi.org/10.1080/17517575.2013.792397

Taylor & Francis
Taylor & Francis Group

Controlled information destruction: the final frontier in preserving information security for every organisation

Daniel-Ioan Curiac* and Mihai Pachia

'Politehnica' University of Timisoara, Automation and Applied Informatics, Bd. V. Parvan nr. 2, Timisoara 300223, Romania

(*Received 16 August 2012; final version received 31 March 2013*)

Information security represents the cornerstone of every data processing system that resides in an organisation's trusted network, implementing all necessary protocols, mechanisms and policies to be one step ahead of possible threats. Starting from the need to strengthen the set of security services, in this article we introduce a new and innovative process named controlled information destruction (CID) that is meant to secure sensitive data that are no longer needed for the organisation's future purposes but would be very damaging if revealed. The disposal of this type of data has to be controlled carefully in order to delete not only the information itself but also all its splinters spread throughout the network, thus denying any possibility of recovering the information after its alleged destruction. This process leads to a modified model of information assurance and also reconfigures the architecture of any information security management system. The scheme we envisioned relies on a reshaped information lifecycle, which reveals the impact of the CID procedure directly upon the information states.

Keywords: controlled information destruction; information security; residual sensitive information; information lifecycle; information security management system

1. Introduction

In any organisation, information security encompasses systems, policies and procedures intended to ensure the confidentiality, integrity and availability of critical data (Furnell 2005). Given the facts that the number and complexity of attacks are growing rapidly, while the security tools are not always handled properly, preserving the privacy of digital data has become a top priority. The scientific research tackled this intricate problem from different perspectives, sometimes using the experiences caught in other security-related fields.

The attempt to implement efficient protocols to destroy digital documents similar to the ones designed for paper documents has put organisations in a tough position. Classic data destruction procedures like media sanitisation (Kissel et al. 2006) address only one facet of this intricate problem by presuming that the information is precisely localised. But this is not a frequent situation. Because of the characteristics of digital data (easy to be shared, copied, modified, transmitted, etc.), a complete disposal is proved to be very difficult, mostly because splinters of information may have already been scattered throughout the organisation's network. In the last decade, the problem is even more severe with the continuous expansion of cloud technologies (Li et al. 2012), requiring an immediate response.

*Corresponding author. Email: daniel.curiac@aut.upt.ro

The information security scenario that conducted to our investigation started with a simple question: what happens when we deal with confidential information that we decide it must be erased? The purpose of this information disposal may vary from the need to free some storage space to the necessity of removing virus-infected data. Our approach focuses on a particular need of the information disposal: the protection of sensitive information that is no longer needed for the organisation's future objectives but would be very detrimental if revealed. Ensuring that this kind of data is completely destroyed can be complicated, mainly because the information in total or in part may be already spread all through the network in different forms (copied, modified, archived, encrypted, etc.) and on diverse devices.

In practice, the most relevant example that proves the necessity to destroy sensitive information immediately after it becomes useless is WikiLeaks (Sifry 2011). Here, thousands of apparently secure and encrypted secret documents were collected by insiders from diverse organisations to be released to an outside party that made them public.

In our view, the controlled disposal of sensitive data must be implemented as a service that can be triggered by an administrator in charge with information security management inside the organisation. We named this service as controlled information destruction (CID).

Besides the information itself (its content), there are some other things we want to protect through CID: the source and the destination of information, the entities that had access to it and even the details about the information lifecycle (e.g. the time when it was created or the moment when destruction was activated).

2. Related work and paper organisation

In the last couple of years, a series of relevant approaches marked the field of securing data through different types of destruction procedures.

In Geambasu et al. (2010), the authors addressed the problem of information self-destruction in such a way that all copies of a certain data become unreadable after a user-specified time period. Their system, named Vanish, is a proof-of-concept prototype that acts in a completely decentralised manner and integrates a novel cryptographic technique with global-scale, P2P, distributed hash tables (DHTs).

A similar strategy, but using a centralised procedure, is described in Perlman (2005). A trusted server called 'ephemeriser' creates temporary keys, makes them available for encryption, helps decryption and destroys the keys at the appropriate time. Ephemeral encryption can be done using either public key scheme or conventional secret-key scheme and must be associated with the use of special software that does not permit to store decrypted messages.

In the wireless sensor networks field, owing to their various military applications, the information destruction plays a major role. In Plastoi and Curiac (2009), the authors describe an efficient power monitoring scheme that triggers a self-destruction procedure to secure sensitive data from battery-powered nodes before they lose contact with the base station. Another approach is revealed by Plastoi et al. (2009), where a self-destructing procedure is started for a cluster-tree wireless sensor network immediately after its operational time expires to protect confidential information.

In the area of hardware technology, there are some relevant implementations that are worth mentioning: (a) a technique for rapid electronic self-destruction of a CMOS integrated circuit, intended to secure the memorised information when discovering an

attempt to examine the chip Shield and Davis (1998); (b) the DS5003 secure microprocessor developed by Maxim, provided with a self-destruct input pin for interfacing to external tamper-detection circuitry that allows an instant removal of the cryptographic key and the deletion of 48-byte vector SRAM area (Maxim Integrated Products 2008); (c) a number of USB flash drives and storage devices (IronKey, Kingston, Victorinox) integrate the data confidentiality protection through self-destruction and (d) the Backstopp technology designed by Virtuity to supply self-destruction features for laptops or smartphones activated when the device leaves a user-defined geographical area or when it is lost or stolen (Virtuity Ltd. 2008).

Our article focuses on the control of information destruction, a concept viewed as a security service. Consequently, this evolution has led to a modified cube-shaped model of information assurance (IA) and to a new view of pursuing information in all its states, based on two interrelated perspectives: from information birth until its death and alongside the path from the source of information toward its destination. By detecting the vulnerabilities and security issues of any information security management systems (ISMS) in terms of sensitive data disposal, we propose an ISMS architecture, which includes the CID service and the related modules.

The rest of this article is organised as follows. In Section 3, we present the CID process with its associated key concepts. Later in this section, we underline the importance of a modified IA model. Section 4 introduces our original information lifecycle perspective that includes the destruction requirements implemented by the CID-related procedure. Section 5 describes the architecture for implementing CID within an ISMS. In section 6, we present a prototype system developed to illustrate our approach, and conclusions being offered in Section 7.

3. Controlled information destruction

Today's informational field is confronted with vast challenges that arise from the need of preserving data security, albeit various combinations of technologies have already been developed. Historically, the construction, maintenance and improvement of information security systems have been difficult. Despite possible problems, any public or private organisation has been struggling to protect sensitive information by abiding to certain rules, carefully and responsibly implementing the required security procedures.

To define CID, a group of related key concepts have to be detailed:

- *Sensitive information*: every kind of data for which the compromise with respect to confidentiality, integrity or availability could have an adverse consequence for the organisation.
- *Residual information*: any useless information or part of information left behind on diverse devices, including information not properly destroyed (information that can be recovered).
- *Information tracking and tracing*: the process of recording every operation (copy, share, modify, transmit, etc.) underwent by information to locate it or its splinters at any moment in time throughout the organisation's network.
- *Complete information destruction*: the process of destroying the information together with all associated data (who and when the information was created, etc.) in such a manner that it cannot be recovered by any means.

Definition: CID represents a specific service that extends the information security beyond information's lifetime by securing through destruction the information itself and its related data (owner, creator, spread in the network, etc.). It is implemented using an assortment of methods, protocols and policies addressed to all sensitive data becoming useless for the organisation that assures their complete destruction, from all locations, leaving no room for residuals.

In our view, CID implementation must be based on the following pillars:

- An information tracking and tracing system
- A strong overall policy for users having access to sensitive information.
- An administrator in charge with triggering the CID related process for any specified sensitive information and in charge with coping with internal/external threats (discovering security leakages). Moreover, this administrator can decide to set a self-destruction process for highly sensitive information that will be triggered automatically anytime its lifetime expires, when it is misused, when it reaches into an unprotected zone or when malicious attacks are encountered
- A special procedure to destroy information, tailored for each and every device

By adding dedicated components, the system can perform additional and more sophisticated tasks, for example tracking and destroying viruses or detecting and stopping insider actions that target important documents.

Besides information disposal, other aspects of CID must be carefully considered: (i) the possibility to include a time-restriction for reading any information or, in other words, to include a time stamp for information self-destruction; (ii) the option to trigger the destruction procedure for defending sensitive data (e.g. tracking and destroying viruses or detecting and stopping insider actions that target important documents, etc.), CID being, to some extent, an assurance that authorised and unauthorised entities allowed to have access to the information will respect other organisational security policies and practices and (iii) the possibility to move information and its splinters to a quarantine zone for a specified period of time before destruction.

As designed, CID will have a major impact on each and every component of the IA model, needing to be integrated into it.

3.1. Towards a new information assurance model

IA was defined by the National Security Telecommunications and Information Systems Security Committee (NSTISSC) as the set of operations that protect and defend information and information systems against possible threats by ensuring their availability, integrity, authentication, confidentiality and non-repudiation (CNSS 2010).

To represent the (information) risk management process, John McCumber (1991) proposed a cube-shaped model (Figure1a) based on three essential aspects:

- *Information states (transmission, storage, processing)*: represent the vast variety of forms in which the information can be met within a system.
- *Countermeasures (technology, policies and practices, people)*: represent diverse actions that can be involved in defending a system from any sorts of attacks against the information
- *Security goals (confidentiality, integrity, availability)*: represent the security services required from any information system.

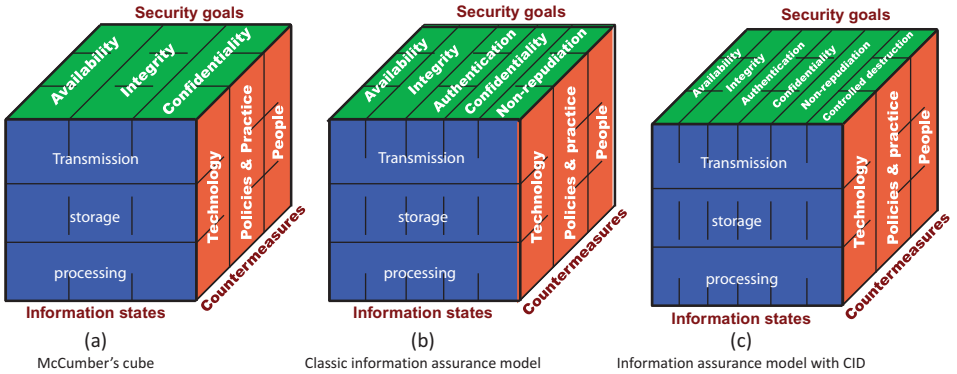

Figure 1. Information assurance models. (a) McCumber's cube. (b) Classic information assurance model. (c) Information assurance model with CID.

The McCumber cube was later expanded in Maconachy et al. (2001) by including authentication and non-repudiation as security services to obtain the IA Model (Figure 1b). In our view, another security goal has to be added to cope with information security during entire information lifecycle and beyond: controlled destruction (Figure 1c).

By including CID, not only that the IA model validity is extended after the formal information lifetime, but also it underlines the need to develop new countermeasures that address this particular service without endangering the other five security objectives. Such an attempt should rely on identifying the states in which information can be found throughout organisational network and on previous knowledge gained from other related fields.

4. CID influence upon the information lifecycle

The concept of controlling the information destruction when it is no longer needed puts its mark on the final stage of a standard information lifecycle. Furthermore, to delete any residual information left behind in different information states and on diverse physical devices, the CID-related procedures have to monitor the entire information lifecycle and its dispersal inside the network.

For a comprehensive analysis of spreading the information or its splinters throughout the organisational network, our approach pursues two interconnected perspectives:

- Throughout the entire lifecycle of the information, from birth until death
- Along its path from the source entity of information to diverse destination entities

These two approaches can be combined by representing the information lifecycle coupled with the transmission process from source to destination, as presented in Figure 2.

Information created during the birth stage develops its lifecycle on each of the entities along its path: source, transmission line and destination. Therefore, the life and death stages can be observed in all the tree entities. Although creating information can be simple, its complete destruction may prove to be much more difficult. This aim can be accomplished only by integrating technologies, policies and trained people in an ensemble

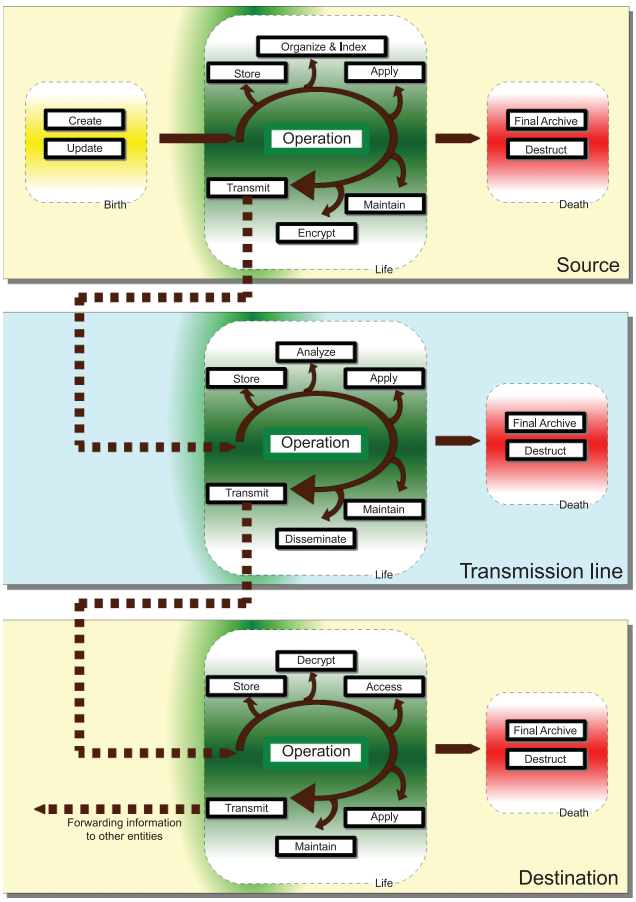

Figure 2. Information lifecycle.

capable to locate and delete information in any particular states of the reshaped information lifecycle presented in Figure 2.

The source entity plays a decisive role in creating, updating and processing the information. Figure 3 illustrates the interactions between different components or operations that affect information on the source level. The CID-related procedures are included in the newly introduced *destruct* component, which controls the information destruction on this level.

As presented in Figure 3, when triggering the destruct component, the information and its splinters localised on the *Store, Final Archive, Organise and Index, Maintain, Apply, Encrypt* and *Transmit* components have to be permanently removed.

The transmission line level signifies an intermediate entity, which establishes the connection between the source and the destination of information and contains essential information states or well-defined interconnected components which operate on the information in transit. Figure 4 illustrates the components of the transmission line level, highlighting that all components are subjected to CID-related procedures covered by *destruct*.

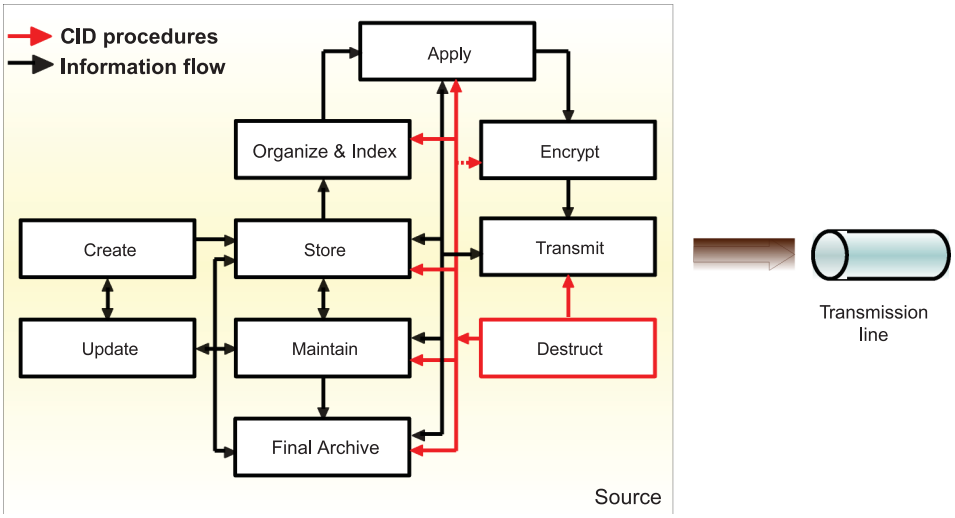

Figure 3. The interaction of information states on the source level.

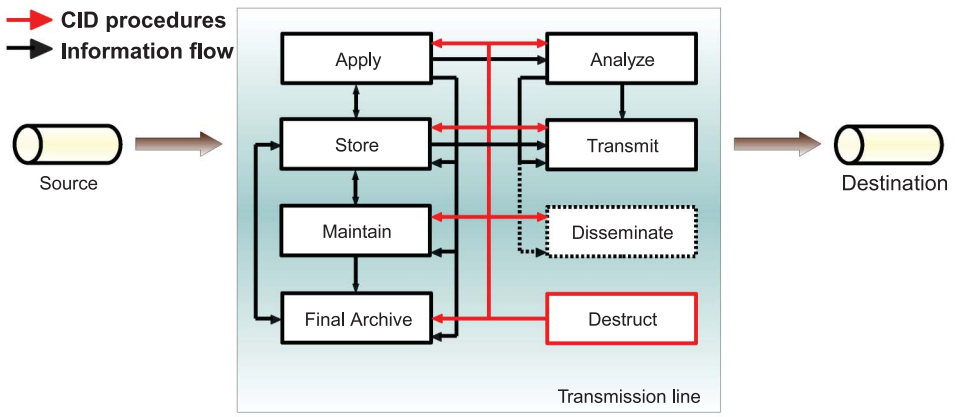

Figure 4. The interaction of information states on the transmission line level.

On transmission line level, the information, sometimes divided in numerous fractions, travels through a lot of intermediary nodes. To delete all information fragments and all data related to information (its source, its path through the network, its destination, time when different operations upon information were encountered, etc.), the CID procedures have to operate on each and every node, covering the entire transmission process.

The destination level represents the last entity involved in the communication chain. This phase encompasses only the components from the life and death stages of the lifecycle (Figure 5).

In many circumstances, the destination decides to forward the information to other entities complicating the CID task by involving new entities in the process.

The analysis of the information lifecycle coupled with information spreading throughout organisational network proves the complexity of the controlled information

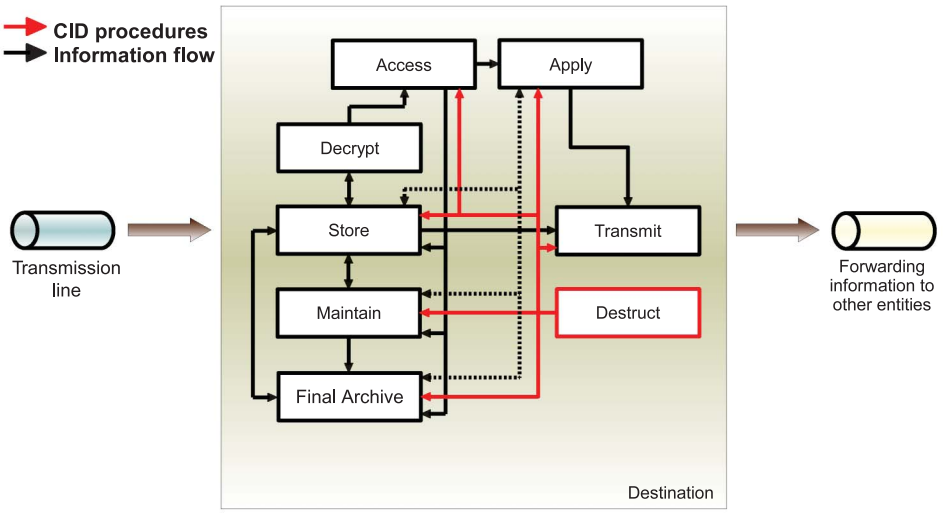

Figure 5. The interaction of information states on the destination level.

destruction procedures and the need to reconfigure the architecture of future information security management systems.

5. Towards an overall architecture of an information security management system including CID

Any organisation aware of the security issues should design, develop and maintain an efficient collection of policies, processes, technologies and systems for administrating the risks to its information assets. Such an assortment of methods can be integrated in a complex scheme, named ISMS (Sanchez et al. 2010; Pereira and Dinis Santos 2010) that copes in a centralised manner with all IT-related risks, assuring the required information security. An ISMS can be viewed as a high-level proactive methodology to continuously and efficiently administer the information security ensemble, including people, infrastructure, technology and businesses (Eloff and Eloff 2003). It must be established in accordance with the ISO/IEC 27000-series (ISMS Family of Standards), which provide a series of recommendations on information security management, risks and controls.

Including the CID procedures in ISMS is related to the necessity of providing the means to trace all relevant information and all related information splinters throughout the network. Additionally, the CID implementation must be flexible and scalable to cover different scenarios and plenty of network-connected devices.

In our view, the development of CID-related procedures is based on the traceability concept, defined in CSA (1994) as the ability to trace the history, application or location of an entity, in our case the sensitive information inside an organisational network, by means of recorded identifications. The traceability is used in both of its forms: traceback (tracing) representing the ability to identify the source of a particular information or splinter; and traceforward (tracking) signifying the ability to pursue the path of an information or splinter of information through the network.

Starting from the traceability concept that governs the entire CID implementation, a set of modules must be included in the standard architecture of ISMS (Wing 2002) as

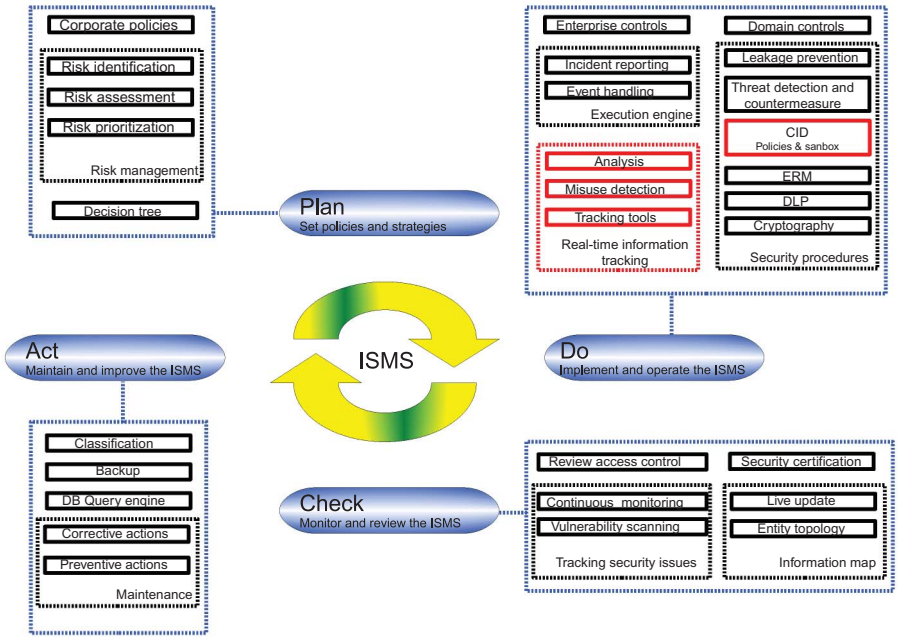

Figure 6. Information security management system including CID additional modules.

presented in Figure 6. Here the components and their interactions are presented based on the Deming Cycle (Plan–Do–Check–Act) (Allen 2008).

5.1. PLAN phase

The first phase of the information security process – *Plan* – includes the design of ISMS in all of its aspects: the analysis of the prevailing circumstances, the assessment of information security risks, the selection of appropriate controls and the establishment of performance evaluation criteria.

Before sketching the ISMS, an organisation must first understand the environment to evaluate the enterprise risks that will be covered by the upper management directives (*Corporate Policies*). The most common areas within an organisation that require the definition of certain policies and procedures are access control, identification and authentication, system maintenance, communication protection, information integrity, physical and environmental protection, audit and accountability, media protection, etc. The set of policies and procedures must be augmented with one related to the need of CID when sensitive information are involved.

The *risk management* section includes the identification, assessment and prioritisation of risks. Risk identification has two directions that need to be investigated before triggering any security procedure: the source of problems (source analysis) and the problem itself (problem analysis). Afterwards, a scale of risk significance is developed for assessment purposes. In the end, a prioritisation process takes place, the risks with the greatest impact (or the greatest probability of occurring) being handled first, while risks with lower probability of occurrence and lower loss are dealt with in a descendant order. As concerning the information destruction process, the *risk management* section has to

identify potential internal and external threats for revealing out-of-use sensitive information and has to prioritise the destruction for every category of confidential data.

The *decision tree* section is designed to efficiently mitigate the encountered risks using appropriate controls based on risk management data.

5.2. DO phase

The ISMS policies, procedures, controls and processes are implemented and operated within the Do phase that is basically an ample state-of-practice guidance illustrating the approach to mitigate against security-related events of diverse types, including the utilisation of antivirus/antispyware software, the configuration of firewalls, the protection of vital servers and subnetworks, the deployment of efficient intrusion detection systems, etc.

The stringent enterprise controls must be implemented according to an information security baseline; these directives being most of the times derived from legislation or industry standards. On this basis, domain controls depend on each unique environment and are derived from the company's internal standards. Therefore, some degree of autonomy is encountered (Carlson 2008).

Every time a security alert is received, the execution engine should be prepared to start automated and nonautomated procedures to follow a well-defined process flow, also monitoring it through all its steps. In the majority of cases, the incident data are immediately added to a log file and reported to the appropriate person. There are two situations in which the execution engine can trigger the CID: (a) when sensitive information is under attack and there is a predefined policy to destruct it prior to be revealed to unauthorised or malicious entities; (b) when time for destruction of out-of-use confidential information arrived.

The CID procedures are based on a complex real-time information tracking module (ITM) that is aimed to locate every sensitive information or its splinters throughout the organisational network at any moment in time. This is done using efficient information tracking tools, which can monitor the spreading of information and the operations upon it. Any encountered event, together with all its attributes, can be held in an event database, while the information about the respective data movement can be kept in a trace repository. Actions of a certain user or entity can easily be traced and examined by the analysis component if it is stored in the event database. The misuse detection component can trigger the CID-related processes according to the levels or severeness of operations anytime unauthorised actions are discovered. When confronting with known or unknown malicious events that sometimes cannot be avoided, the system's security services must rely on sandboxes to study the behaviour of attackers and to annihilate them without other further risks.

Inaccurate and ineffective controls can lead to countless circumstances in which security services are jeopardised, requiring efficient security procedures that include the use of cryptography and tools for leakage prevention or threat detection and countermeasure. Enterprise rights management (ERM) (Gasmi et al. 2008) represents a security technology that institutes certain rights over document usage and also constantly encrypts data. At present, common IT tools use embedded encryption technologies to protect sensitive information within applications (Whitman and Mattord 2007). The presence of sensitive information tracking inside ISMSs have pointed out that there are several approaches that lead to the successful monitoring of information in unsafe environments. Data loss prevention technologies (Lesnykh 2011) discover unauthorised usage of

information, also monitoring and protecting confidential data in use (e.g. endpoints), data in motion (e.g. network actions) and data at rest (e.g. data storage).

An important issue for any ISMS is related to the defending actions against insider threats. Discovering internal attackers will always be difficult because they can access the system through a lot of ways based on inside information, trying to hide their malicious intentions by not drawing attention on them. The potential damage can be mitigated by creating operation hierarchies that help analysts define specific levels of risk, furthermore applying the best course of action (You et al. 2012).

5.3. *CHECK phase*

The system and its security performance against various pre-established specifications or standards are evaluated on a regular and periodic basis in the *check* phase of ISMS by performing a set of operations like monitoring, measuring, reviewing or assessing.

The *check* phase incorporates the examination of all security alert mechanisms including the observation of system logs, firewall logs or intrusion detection logs. Moreover, it may include accomplishing and reviewing the effects of vulnerability assessments, penetration tests, security risk assessments, IT audits and processes.

The implemented system must be well-mapped out and documented (an *information map* must be kept and updated) so that an authorised entity, with administrative rights, can do the following:

- Confidently query any entry from the central database
- Compare changes between document versions (e.g. time and date an item was moved)
- Retrieve relevant documentation (sometimes reviewing previously saved logs)
- Find out if sufficient metadata exists to support any changes in parameter values

Live update guarantees that data are mapped to the exact entity or person the instant an action occurs.

Ideally, a system security plan should be developed before the security certification process of the ISMS. System security plans require periodic review, adjustment and defined deadlines for implementing security controls (Swanson, Hash, and Bowen 2006).

5.4. *ACT phase*

The final phase – *Act* – has the purpose of maintaining and improving the ISMS. It is often initiated as a reactive task to remediate inadequate controls or to countermeasure security gaps that were detected. When the system becomes stable, resources can converge to more proactive and preventive measures. The corrective and preventive actions must be based upon internal auditing – testing might be involved – and sometimes upon higher management reviews. The continuous improvement of the ISMS, including the CID-related processes, can be achieved through a plan of action based on the results of the preceding stages to address any shortcomings found. As a consequence, the cycle of continuous enhancements is carried out indefinitely by proposing new improvements, updating the risk assessment, upgrading procedures and controls, requiring additional resources or by enriching the ways of evaluating the effectiveness of existing policies.

Managing sensitive information is directed by a variety of data entry tools, operators being forced to follow the correct process flow. Methods are documented by procedures that check by whom, whence, when and how information was accessed.

6. System implementation

A prototype software system – NetInfoShred, accompanied by a well-suited set of policies – was implemented to illustrate the proposed approach. The starting point was the following basic scenario: (a) a file marked as 'possible confidential file' (PCF) is placed somewhere in the LAN by the security administrator; (b) all accesses to this file together with related events (change file, delete file, rename file, etc.) are carefully monitored and stored in a database. Moreover, every new file containing parts of information from initial PCF file has also to be marked as PCF; (c) when the security administrator decides, a batch program is launched to delete the file together with all its copies and all its splinters. Actually, NetInfoShred includes many more features like the possibility to run scheduled automatic deletion batches or to monitor the violations of policies. The architecture of NetInfoShred includes a group of modules presented below:

- *Information tracking module*: is permanently monitoring the activity on each and every computer to discover the way information is used and stores the data in an SQL database. This module, developed in C#, is based on FileSystemWatcher class (Microsoft 2012), which raises the events described in Table 1.
- *Information splinters discovery module (ISDM)*: is practically a search engine that discovers parts of different documents copied in other files on the LAN and works in conjunction with ITM or as a standalone module. Its C# code is based on the key idea of automated text-matching screening used by anti-plagiarism systems (Li 2012; Lukashenko, Graudina, and Grundspenkis 2007; Maurer, Kappe, and Zaka, 2006), which involves a sliding window technique to search for sequences of words from a target file in other files located on the LAN.
- *Deletion batch module (DBM)*: is a C# program that can delete and verify the deletion process of every file in the LAN that is associated with specified confidential information. It can be scheduled by the security administrator for automated deletion at a particular moment in time or can be used to delete the files one by one, in a supervised approach.
- *Policy violation alerts module (PVAM)*: is addressing a collateral problem that unquestionably represents a potential threat for every confidential file: violations

Table 1. Events monitored using FileSystemWatcher class.

Event name	Description
Changed	Occurs when a file or directory in the specified Path is changed.
Created	Occurs when a file or directory in the specified Path is created.
Deleted	Occurs when a file or directory in the specified Path is deleted.
Disposed	Occurs when the component is disposed by a call to the Dispose method. (Inherited from Component.)
Error	Occurs when the instance of FileSystemWatcher is unable to continue monitoring changes or when the internal buffer overflows.
Renamed	Occurs when a file or directory in the specified Path is renamed.

of policies. The set of policies attached to NetInfoShred prevent the occurrence of circumstances in which splinters of the confidential file may escape outside ITM/ISDM coverage. Some relevant examples include the use of clipboard copy/paste method, the use of PrintScreen key or screen capture software, saving files on portable storage devices, etc. In these cases, we decided to automatically monitor violations of policies and to trace them in the database to be analysed by the security administrator.

- *Administrator dashboard module (ADM)*: is a php/javascript module that allows the security administrator to set and view the entire process in real time. It also includes an alert system that sends SMS or email messages when certain situations occur and various reporting or dashboard tools. Figure 7 shows a snapshot of this visual module.

All events or activities that can produce information splinters must either be covered by automated information tracking/discovering methods or forbidden by precise policies. Although the automated mechanisms have high reliability, banning activities through policies is often subject to human error, misinterpretation or even malicious actions. For this reason, it is our primary goal to minimise the number of needed policies by raising the coverage of automated mechanisms. Therefore, we have designed two new special metrics to evaluate our software development process – percentage of banned events without related alerts (PBEWA) imposed through policies and percentage of monitored events (PME):

$$\text{PBEWA} = \frac{\text{Number of events banned through policies without alerts}}{\text{Total number of events}} \times 100 \tag{1}$$

$$\text{PME} = \frac{\text{Number of events automatically monitored}}{\text{Total number of events}} \times 100 = 100 - \text{PBEWA} \tag{2}$$

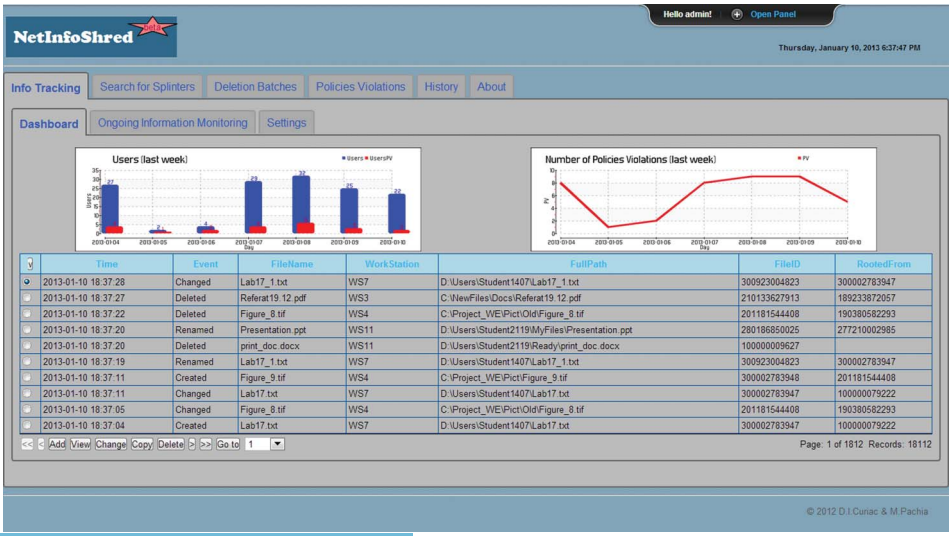

Figure 7. Snapshot of the security administrator's dashboard.

Table 2. Implementation sequence.

Stage	Splinters tracking modules	Set of policies with implemented alerts (PVAM constituents)	PBEWA (%)
0	–	–	~100
1	ISDM	–	64
2	ISDM + IDM	–	24
3	ISDM + IDM + PVAM	P0 + P1	13
4	ISDM + IDM + PVAM	P0 + P1 + P2	9
5	ISDM + IDM + PVAM	P0 + P1 + P2 + P3	3
6	ISDM + IDM + PVAM	P0 + P1 + P2 + P3 + P4	1

While PBEWA must be decreased towards zero, its opposite metric (PME) must rise up towards 100%.

A local area network, with an Internet connection, composed of 12 computers running Windows 7 (one is acting as a server) represented the experimental environment used to evaluate our approach. The implemented software modules were included on the test platform one-by-one to measure their impact upon PBEWA. The stage-by-stage implementation process is presented in Table 2, where the policies with implemented alerts are denoted by the following: P0 – The use of portable storage devices (e.g. memory cards, CDs/DVDs, etc.) is forbidden during access to a PCF file; P1 – The use of clipboard, including PrintScreen key, is forbidden during access to a PCF file; P2 – The use of other programs in parallel with the one used for accessing the 'PCF' is forbidden during access to a PCF file; P3 – Any external connection to devices outside LAN (through the internet, Bluetooth, etc.) is forbidden during access to a PCF file; and P4 – The use of file compression or file encryption is forbidden during access to a PCF file.

In the preliminary stage (stage 0), we evaluated, based on probabilistic analysis, the way information or splinters of information may be spread inside/outside the LAN. In this stage, the set of policies that prevents possible spreading of information had to address almost 100% of the situations. The only allowed circumstance, not banned by policies in stage 0, was the one when the PCF marked file is modified and the new file version replaces the former file (is placed in the same location). In the following stages, subsets of policies were replaced by automatic mechanisms. A graphical coverage of splinter-related events in every implementation stage is presented in Figure 8, while the variation of the PBEWA and PME metrics are depicted in Figure 9.

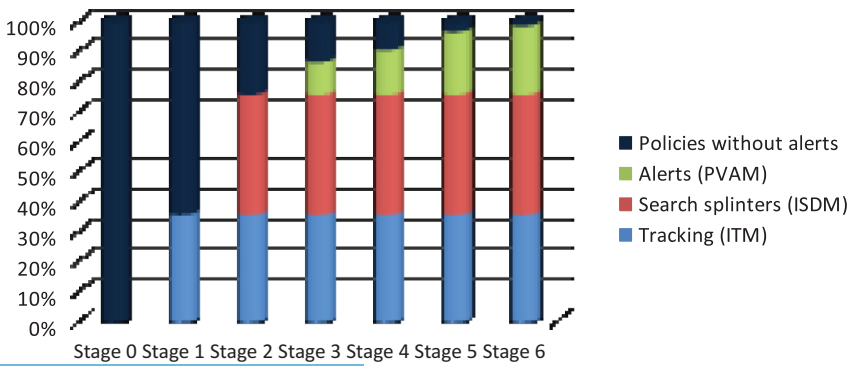

Figure 8. Splinter-related events coverage during software implementation process.

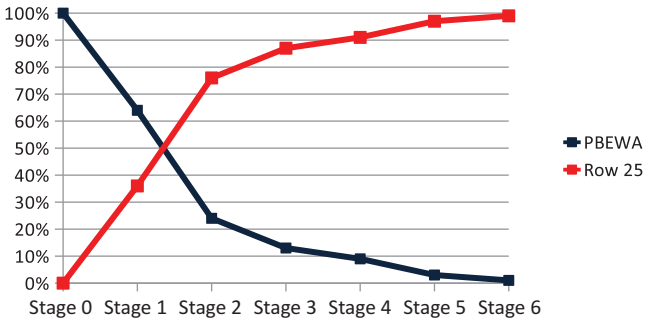

Figure 9. PBEWA and PME metrics variation during software implementation process.

In the development and testing process of the prototype software system, a set of learned lessons is worth being highlighted:

- It is obvious that the most demanding part of the entire CID process is related to tracking the information spreading throughout the network. This cannot be done using a single method, but with a mixture of methods and policies that work synergically.
- CID closely interacts with two important software entities: network operating system and antivirus programs. A simultaneous and combined design of these three components may bring a higher reliability to the entire system.
- The design of well suited policies attached to CID must not rely on user anticipated behaviour, but on banning all possible activities that cannot be automatically monitored.

7. Conclusions

Starting from the need to secure confidential information that is no longer needed for an organisation but would be very detrimental if revealed, we designed a new security service to cope with this problem. This service named CID extends the limits of the information security beyond the formal death of any sensitive information using a radical procedure: erasing the information together with all related attributes from every device inside an organisational network, leaving no room for residuals and no possibility for information recovery. To design CID-related procedures, we pointed out the need to rely on efficient information tracking systems that can pursue the information and its splinters through all phases of information lifecycle and on the road between the source of information and its destination. Knowing the possible location of information splinters throughout the organisational network, an information destruction procedure, tailored for every encountered device, must be started.

By including this novel service in an information security management system, the security level will be increased significantly. Moreover, we underlined that CID and the related procedures and policies can extend their role representing a final countermeasure when internal or external attackers are on the edge to access classified information or when coping with malicious viruses that penetrated the system.

References

Allen, J. H. 2008. "Plan, Do, Check, Act." Build Security Accessed August 14. https://buildsecurityin.us-cert.gov/bsi/articles/best-practices/deployment/574-BSI.html

Carlson, T. 2008. "Understanding Information Security Management Systems." In *Information Security Management Handbook*. 6th ed., Vol. 2, ed. H. F. Tipton and M. Krause, 15–28. vol Boca Raton, FL: Auerbach Publications.

CNSS (Committee on National Security Systems). 2010. National Information Assurance (IA) Glossary, CNSSI 4009, 35. April 26.

CSA (Canadian Standards Association). 1994. *ISO 8402:1994, Quality management and quality assurance – Vocabulary*. Rexdale, Toronto, ON: Canadian Standards Association.

Eloff, J., and M. Eloff. 2003. "Information Security Management: A New Paradigm." In *Proceedings of the 2003 Annual Research Conference of the South African Institute of Computer Scientists and Information Technologists on Enablement through Technology*, 130–136, September 17–19. Fourways: SAICSIT Press.

Furnell, S. 2005. *Computer Insecurity – Risking the System*. New York: Springer-Verlag.

Gasmi, Y., A. R. Sadeghi, P. Stewin, M. Unger, M. Winandy, R. Husseiki, and C. Stüble. 2008. "Flexible and Secure Enterprise Rights Management Based on Trusted Virtual Domains." In *Proceedings of the 3rd ACM Workshop on Scalable Trusted Computing*, 71–80, October 27–31. Alexandria, VA: ACM.

Geambasu, R., T. Kohno, A. A. Levy, and H. M. Levy. 2010. "Vanish: Increasing Data Privacy with Self-Destructing Data." In *Proceedings of the 9th USENIX Symposium on Operating Systems Design and Implementation (OSDI)*, 299–315, October 4–6. Berkeley, CA: USENIX Association.

Kissel, R., M. Scholl, S. Skolochenko, and X. Li. 2006. "Guidelines for Media Sanitization, Recommendations of the National Institute of Standards and Technology." *NIST Special Publication 800-88*.

Lesnykh, A. 2011. "Data Loss Prevention: A Matter of Discipline." *Network Security* 2011, 3: 18–19.

Li, S., L. Xu, X. Wang, and J. Wang. 2012. "Integration of Hybrid Wireless Networks in Cloud Services Oriented Enterprise Information Systems." *Enterprise Information System* 6, 2: 165–187.

Li, Y. 2012. "Text-Based Plagiarism in Scientific Publishing: Issues." *Developments and Education*, *Science and Engineering Ethics*, 1–14. doi: 10.1007/s11948-012-9367-6.

Lukashenko, R., V. Graudina, and J. Grundspenkis. 2007. "Computer-Based Plagiarism Detection Methods and Tools: An Overview." In *Proceedings of the 2007 International Conference on Computer Systems and Technologies (CompSysTech'07)*, June 14–15, Article No. 40. Rousse: ACM.

Maconachy, W. V., C. D. Schou, D. Ragsdale, and D. Welch. 2001. "A Model for Information Assurance: An Integrated Approach." In *Proceedings of the 2001 IEEE Workshop on Information Assurance and Security*, 306–310, June 5–6. West Point, NY: US Military Academy.

Maurer, H., F. Kappe, and B. Zaka. 2006. "Plagiarism – A Survey." *Journal of Universal Computer Sciences* 12, 8: 1050–1084.

Maxim Integrated Products. 2008. *DS5003 Secure Microprocessor Chip Datasheet*, Maxim Integrated Products. Accessed August 14. http://datasheets.maxim-ic.com/en/ds/DS5003.pdf

McCumber, J. 1991. "Information Systems Security: A Comprehensive Model." In *Proceedings of the 14th NIST-NCSC National Computer Security Conference*, 328–337, October 11–14. Baltimore, MD: NIST.

Microsoft. 2012. "FileSystemWatcher Class." MSDN/System.IO. Accessed August 14. http://msdn.microsoft.com/en-us/library/system.io.filesystemwatcher.aspx

Pereira, T., and H. Dinis Santos. 2010. "An Audit Framework to Support Information System Security Management." *International Journal of Electronic Security and Digital Forensics* 3, 3: 265–277.

Perlman, R. 2005. "The Ephemerizer: Making Data Disappear." *Information System Security* 1 (1): 51–68.

Plastoi, M., and D.-I. Curiac. 2009. "Energy-Driven Methodology for Node Self-Destruction in Wireless Sensor Networks." In *Proceedings of the 5th International Symposium on Applied Computational Intelligence and Informatics (SACI'09)*, 319–322, May 28–29. Timisoara: IEEE.

Plastoi, M., D.-I. Curiac, O. Banias, C. Volosencu, D. Pescaru, and A. Doboli. 2009. "Self-Destruction Procedure for Cluster-tree Wireless Sensor Networks." In *Proceedings of the International Conference on Wireless Information Networks and Systems (WINSYS 2009)*, 63–67, July 7–10. Milan: INSTICC Press.

Sanchez, L. E., A. Santos-Olmo, E. Fernandez-Medina, and M. Piattini. 2010. "Building ISMS through the Reuse of Knowledge." In *Proceedings of the 7th International Conference on Trust, Privacy and Security in Digital Business*, 190–201, August 30–31. Bilbao: Springer.

Shield, J. D., and L. D. Davis. 1998. Method and apparatus for fast self-destruction of a CMOS integrated circuit. US Patent 515 5736777, April 7.

Sifry, M. L. 2011. *WikiLeaks and the Age of Transparency*. Berkeley, CA: Counterpoint Press.

Swanson, M., J. Hash, and P. Bowen. 2006. "Guide for Developing Security Plans for Federal Information Systems." *NIST Special Publication 800-18*, Revision 1.

Virtuity Ltd. 2008. *What is Backstopp?* Backstopp, Virtuity Ltd., Accessed August 14. http://www.backstopp.com/what_is_backstopp.aspx

Whitman, M. E., and J. Mattord. 2007. *Principles of Information Security*. 3rd ed. Boston, MA: Thomson Course Technology.

Wing, R. 2002. *SP-018: Information Security Management System (ISMS) Module*. Open Security Architecture. Accessed August 14. http://www.opensecurityarchitecture.org/cms/en/library/patternlandscape/279-module-information-security-management-system-isms

You, I., G. Lenzini, M. Ogiela, and E. Bertino. 2012. "Defending Against Insider Threats and Internal Data Leakage." *Security and Communication Networks* 5, 8: 831–833.