Mise en oeuvre de la SSI (Sécurité du Système d'information) de SUSS MicroOptics par l'approche processus ISO/CEI 27001

Thierry Boileau

CONSERVATOIRE NATIONAL DES ARTS ET METIERS

CENTRE REGIONAL ASSOCIE DE LYON

_____

MEMOIRE

présenté en vue d'obtenir

le DIPLOME D'INGENIEUR CNAM

SPECIALITE : INFORMATIQUE, SYSTEME D'INFORMATION

Par

Thierry BOILEAU

_____

Mise en oeuvre de la SSI (Sécurité du Système d'information) de SUSS MicroOptics par l'approche processus ISO/CEI 27001

Soutenu le 16 septembre 2010

_____

JURY

PRESIDENT :

M. Christophe Picouleau (CNAM Paris)

MEMBRES :

M. Bertrand David (CNAM Lyon)

M. Claude Genier (CNAM St Genis Pouilly)

M. Jean Marc Valentin (Efficacité.ch)

M. Reinhard Völkel (Suss MicroOptics)

REMERCIEMENTS

Ce mémoire est l'aboutissement d'une série de cours suivis au CNAM. Aussi, je remercie tous mes professeurs ainsi que le personnel administratif pour leur dévouement pour les élèves de cette institution.

J'exprime également mes remerciements à M. Claude Genier, pour ses conseils et le suivi des anciens élèves qu'il n'a eu de cesse d'encourager à présenter un mémoire.

Je remercie, M. Reinhard Völkel, directeur de Suss MicroOptics, qui a adhéré sans condition à l'idée de la rédaction de ce mémoire sur un projet que nous avions en commun. M. Reinhard Völkel a mis à disposition toutes les ressources nécessaires pour que ce travail puisse se faire dans les meilleures conditions. Je remercie les collaborateurs de Suss MicroOptics pour leur disponibilité et tout particulièrement M. Jocelyn Berger qui a beaucoup été sollicité.

Je remercie également M. Jean Marc Valentin, consultant et lead auditeur des systèmes de management de la sécurité de l'information qui m'a guidé tout au long de ce travail.

Enfin, je remercie ma famille pour son soutien.

LISTE DES ABREVIATIONS

AESC : American Engineering Standards Committee

AFNOR : Association Française de Normalisation

ANSSI : Agence Nationale de la Sécurité des Systèmes d'Information

ASA : American Standards Association

ASN.1 : Abstract Syntax Notation One

BSI : British Standards Institute

CA : Chiffre d'Affaire

CDD : Charge-Coupled Device

CEI : Commission Electronique Internationale

CEO : Chief Executive Officer

CEN : Comité Européen de Normalisation

CGI : Common Gateway Interface

CLUSIF : Club de la Sécurité de l'Information Français

COBIT : Control Objectives for Information and related Technology

COO : Chief Operating Officer

CMIP : Common Management Information Protocol

CMIS : Common Management Information Services

CMMI : Capability Maturity Model Integration

CPU : Central Processing Unit

CSEM : Centre Suisse d'Electronique et de Microtechnique

CTO : Chief Technology Officer

DC : Contrôleur de Domaine

DNS : Nom de Serveur de Domaine

DHCP : Dynamic Host Configuration Protocol

DRP : Data Recovery Plan

EBIOS : Etude des Besoins et Identification des Objectifs de Sécurité

ENISA : European Network and Information Security Agency

GCG : GNU Compiler Collection

GD : Gift Draw

GNU/GPL : General Public License

HTML : Hyper Text Markup Language

IMAP : Internet Message Access Protocol

HTTP : HyperText Transfer Protocol

IETF : Internet Engineering Task Force

IMT : Institut de Microtechnique de Neuchâtel

ISO : Organisation Internationale de Normalisation

JTC : Joint Technical Committee

LAN : Local Area Network

MARION : Méthode d'Analyse de Risques Informatiques Optimisée par Niveau

MDA : Mail Delivery Agent

MEHARI : Méthode Harmonisée d'Analyse des Risques

MELISA : Méthode d'Evaluation de la Vulnérabilité Résiduelle des Systèmes d'Armement

MIB : Management Information Base

MOEMS : Microsystèmes Opto-Electro-Mécaniques

MRTG : Multi Router Traffic Grapher

MTA : Mail Transport Agent

MUA : Mail User Agent

NMS : Network Management Station

NSCA : Nagios Service check Acceptor

NSClient : Nagios Service Client

NRPE : Nagios Remote Plugin Executor

OCTAVE : Operationally Critical Threat, Asset, and Vulnerability Evaluation

OID : Object Indentifier

OSI : Open Systems Interconnection

PDCA : Plan, Do, Check, Act

PHP : Hypertext Preprocessor

PME : Petites et Moyennes Entreprises

POP : Post Office Protocol

RAM : Random Access Memory

RATP : Risk Assessment Treatment Plan

R&D : Recherche et Développement

RDDTool : Round-Robin Database Tool

RFC : Request For Comments

SC : Sub-Committees

SGMP : Simple Gateway Monitoring Protocol

SoA : Statement of Applicability

SoM : Statement of Maturity

SMO : Suss MicroOptics

SMSI : Système de Management de la Sécurité de l'Information

SMTP : Simple Mail Transfer Protocol

SMS : Short Message Service

SNMP : Simple Network Management Protocol

SSL : Secure Sockets Layer

TCP/IP : Transmission Control Protocol, Internet Protocol

TI : Technologies de l'Information

UDP : User Datagram Protocol

VPN : Virtual Private Network

W3C : World Wide Web Consortium

WD : Working Draft

WG : Working Group

TABLE DES MATIERES

INTRODUCTION

L'objet de ce mémoire est de montrer l'apport d'un système de management de la sécurité de l'information à travers la mise en œuvre d'une plate-forme de supervision.

L'omniprésence numérique croissante soutenue par les réseaux informatiques en constante évolution, a amplifié le besoin d'assurer l'intégrité, la confidentialité et la disponibilité de l'information.

Malgré des moyens techniques permettant de contrôler en temps réel la qualité et l'état d'un réseau et des compétences pour les mettre en œuvre, la complexité des systèmes d'information exige une planification rigoureuse de la supervision.

Des outils sous forme de « codes de bonnes pratiques » et méthodes d'appréciation des risques permettent aux entreprises de fixer des objectifs, des priorités et coordonner les actions à entreprendre.

Néanmoins ces outils, bien qu'ayant prouvé leur efficacité, souffrent d'un manque de reconnaissance au plan international à cause de leur trop grande diversité.

Pour remédier à l'hétérogénéité des méthodes et pallier le manque de reconnaissance des « codes de bonnes pratiques », l'ISO (Organisation Internationale de Normalisation) a publié en 2005 la norme ISO/CEI 27001.

L'ISO/CEI 27001, clef de voûte d'une famille de normes encore en développement, apporte une dimension supplémentaire à la politique de sécurité de l'information en y intégrant le concept de « système de management ». Cette norme est à la base de notre réflexion pour ce mémoire.

Dans une première partie, nous présentons un état de l'art des systèmes de management de la sécurité de l'information.

Dans une deuxième partie, nous détaillons la norme ISO/CEI 27001 et son approche en quatre phases (Plan, Do, Check, Act).

Dans une troisième partie, nous évoquons à travers un choix d'exemples, l'utilisation de cette norme dans une PME spécialisée en micro-optique. Cette analyse nous permet d'identifier les risques nécessitant l'application de mesures de sécurité.

Une des mesures de sécurité est la supervision. Nous définissons ses concepts et ses notions ainsi que l'implémentation qui en a été faite dans la quatrième partie.

PARTIE I. ETAT DE L'ART DE LA SECURITE DE L'INFORMATION

L'objectif de cette première partie est de présenter les normes, les concepts, les processus et les acteurs qui ont permis aux organismes[1] d'aboutir à un système de management de la sécurité de l'information.

1. Définitions

1.1 L'ISO (Organisation Internationale de Normalisation)

L'ISO est le fruit d'une collaboration entre différents organismes de normalisation nationaux. Au début du XXième siècle, L'American Institute of Electrical Engineer[2] invite quatre autres instituts professionnels pour constituer une première organisation nationale, l'AESC (American Engineering Standards Committee) qui aura pour objectif de publier des standards[3] industriels communs avant de prendre le nom d'ASA (American Standards Association) et d'établir des procédures standardisées pour la production militaire pendant la seconde guerre mondiale. En 1947, l'ASA, le BSI (British Standards Institute), l'AFNOR (Association Française de Normalisation) et les organisations de normalisation de 22 autres pays fondent l'Organisation Internationale de Normalisation (ISO).

A ce jour, l'ISO regroupe 157 pays membres, et coopère avec les autres organismes de normalisation comme le CEN (Comité européen de normalisation) ou la Commission Electronique Internationale[4] (CEI). En 1987, l'ISO et le CEI créent le Joint Technical Committee (JTC1) pour la normalisation des Technologies de l'Information (TI). Le JTC1 allie les compétences de l'ISO en matière de langage de programmation et codage de l'information avec celles du CEI qui traitent du matériel tel que les microprocesseurs. Le JTC1 est composé

[1] Organisme est le terme générique employé pour définir l'entreprise, nous l'utiliserons systématiquement dans ce document. L'ISO définit l'organisme comme l'ensemble des installations et des personnes avec des responsabilités, des pouvoirs et des relations.

[2] Aujourd'hui appelé Institute of Electrical and Electronics Engineers ou IEEE avec comme objectif la promotion de la connaissance dans le domaine de l'ingénierie électrique.

[3] On parle de standard quand le référentiel (document) est diffusé librement comme c'est le cas par exemple pour les recommandations du W3C ou le PostScript publié d'Adobe.

[4] CEI est chargée de la normalisation d'équipements électriques. Il est courant de voir ISO/CEI pour nommer une norme élaborée conjointement par les deux organismes.

de plusieurs comités techniques (SC) qui traitent de sujets tels que la biométrie, la téléinformatique, les interfaces utilisateurs ou encore les techniques de sécurité de l'information relatives aux normes de la série ISO/CEI 2700x. La figure 1 ci-dessous montre la structure hiérarchique des différents groupes de travail tel que le WG1[5] issu du JTC1/SC27 de l'ISO/CEI [1].

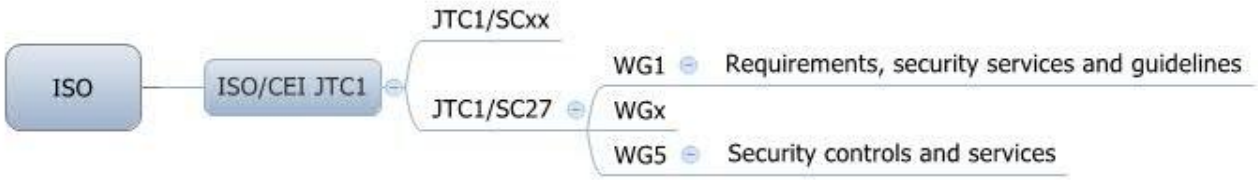


Figure 1 Structure hiérarchique des groupes de travail et comités de l'ISO/CEI

Créé en 2006, le JTC1/SC27 de l'ISO/CEI a développé un nombre important de normes au sein du WG1, celles de la famille ISO/CEI 2700x.

1.2 Les normes

L'ISO et le CEI donnent la définition suivante : «(…) *document établi par consensus et approuvé par un organisme reconnu, qui fournit, pour des usages communs et répétés, des règles, des lignes directrices ou des caractéristiques, pour des activités ou leurs résultats garantissant un niveau d'ordre optimal dans un contexte donné*[6]. » Les documents établis par consensus sont appelés « norme ». On distingue quatre familles de normes:

- Les normes fondamentales concernent les règles qui ont trait à la métrologie.
- Les normes de spécifications, traitent des caractéristiques et des seuils de performance d'un produit ou d'un service.

[5] Le WG1 est le groupe de travail en charge d'organiser et rédiger les normes liées au domaine de la sécurité de l'information.

[6] Directives ISO/CEI – partie 2 : Règles de structure et de rédaction des Nomes internationales, cinquième édition, 2004 (§ 3.1).

- Les normes d'analyse et d'essais, renseignent sur les méthodes et moyens pour la réalisation d'un essai sur un produit.

- Les normes d'organisation qui décrivent les fonctions et les relations organisationnelles au sein d'un organisme.

Ces normes peuvent relever de groupes ou d'organismes de normalisation internationaux ou nationaux tels que le CEN qui a pour but l'harmonisation des normes nationales européennes, l'ISO qui publie les normes internationales et L'AFNOR qui s'occupe des normes françaises [2].

1.3 La normalisation

Selon le Journal Officiel du gouvernement français, la normalisation «(…)*a pour objet de fournir des documents de référence élaborés de manière consensuelle par toutes les parties intéressées, portant sur des règles, des caractéristiques, des recommandations ou des exemples de bonnes pratiques relatives à des produits, à des services, à des méthodes, à des processus ou à des organisations. Elle vise à encourager le développement économique et l'innovation tout en prenant en compte des objectifs de développement durable* (…) » (Art. 1 du décret du 16 juin 2009)[7]. Les documents de référence élaborés au terme du processus de normalisation sont les « normes »[8].

1.4 Historique des normes en matière de sécurité de l'information

Au cours des vingt dernières années les normes liées à la sécurité de l'information ont évolué ou ont été remplacées. Ces changements rendent difficile une bonne compréhension du sujet. Un rappel historique de l'évolution de ces normes permet de clarifier la situation normative en matière de sécurité de l'information.

---

[7] Le Décret n° 2009-697 du 16 juin 2009 relatif à la normalisation, JO du 17 juin 2009, explicite le fonctionnement du système français de normalisation et rappelle la procédure d'élaboration et d'homologation des projets de normes et les modalités d'application des normes homologuées.

[8] Un logigramme du processus d'élaboration des normes suivant les directives de l'ISO/CEI figure en Annexe 1.

Au début des années 90, de grandes entreprises britanniques se concertent pour établir des mesures visant à sécuriser leurs échanges commerciaux en ligne. Le résultat de cette collaboration servit de référence en la matière pour d'autres entreprises qui souhaitaient mettre en œuvre ces mesures. Cette initiative privée fut appuyée par le Département des Transports et de l'Industrie britannique qui supervisa la rédaction au format du BSI, d'une première version de projet de norme de gestion de la sécurité de l'information.

En 1991, un projet de « best practices » code de bonnes pratiques, préconise la formalisation d'une politique de sécurité de l'information. Cette politique de sécurité doit intégrer au minimum huit points « stratégique et opérationnel[9] » ainsi qu'une mise à jour régulière de la politique.

En 1995, le BSI publie la norme BS7799 qui intègre dix chapitres réunissant plus de 100 mesures détaillées de sécurité de l'information, potentiellement applicables selon l'organisme concerné.

En 1998, la norme BS7799 change de numérotation et devient la norme BS7799-1. Elle est complétée par la norme BS7799-2 qui précise les exigences auxquelles doit répondre un organisme pour mettre en place une politique de sécurité de l'information. Cette nouvelle norme est fondée sur une approche de la maîtrise des risques et sur le principe du management de la sécurité de l'information.

En 2000, la norme BS7799-1, devient la norme de référence internationale pour les organismes souhaitant renforcer leur sécurité de l'information. Après avoir suivi un processus de concertation au niveau international et quelques ajouts, l'ISO lui attribue un nouveau nom, ISO/IEC 17799 :2000.

En 2002, le BSI fait évoluer la norme BS7799-2 en s'inspirant des normes ISO 9001 :2000 et ISO 14001 :1996. La norme adopte définitivement une approche de management de la sécurité de l'information.

[9] Les points « opérationnel » de la politique de sécurité de l'information peuvent par exemple concerner la politique de sauvegarde, des mots de passe. Les points « stratégiques » concerneront les engagements de la direction vis-à-vis de la sécurité de l'information.

En 2005, l'ISO/CEI adopte la norme BS7799-2 sous la référence ISO/CEI 27001 :2005 en y apportant quelques modifications pour se rapprocher le plus possible du principe de « système de management » développé par les normes ISO 9001 et ISO14001. L'ISO/IEC 27001 :2005 spécifie les exigences pour la mise en place d'un SMSI (système de management de l'information).

En 2007, dans un souci de clarification, l'ISO renomme la norme ISO/IEC 17799 :2005 en changeant sa numérotation pour ISO/IEC 27002. La norme se greffe à la famille des normes ISO/IEC 2700x toujours en développement.

La figure 2 ci-dessous résume l'historique des normes traitant de la sécurité de l'information.

Figure 2 Historique des normes liées à la sécurité de l'information

Aujourd'hui les organismes disposent de deux normes qui se sont imposées comme référence des SMSI, l'ISO/CEI 27001 :2005 qui décrit les exigences pour la mise en place d'un SMSI et l'ISO/CEI 27002 qui regroupe un ensemble de bonnes pratiques « best practices » pour la gestion de la sécurité de l'information. Autour de ces deux normes viennent s'articuler d'autres normes de la même famille, ISO/CEI 2700x, encore en développement pour certaines [3].

Dans la partie qui suit nous présentons les principales propriétés d'un SMSI avant d'aborder les normes de la série ISO/CEI 2700x qui se sont imposées comme références des SMSI.

2. Les SMSI (Systèmes de Management de la Sécurité de l'Information)

2.1 Les systèmes de management

La norme ISO 9000 définit le système de management comme : *(…) un système permettant d'établir une politique, des objectifs et atteindre ces objectifs (…).*

Un système de management peut être interprété comme un ensemble de mesures organisationnelles et techniques ciblant un objectif comme le montre la figure 3 ci-dessous.

Figure 3 Vue d'un système de management

Un système de management se caractérise par un engagement de l'ensemble des collaborateurs de l'organisme ; quel que soit le périmètre du système sur l'activité de l'organisme, il nécessite l'implication de tous les métiers. A cette approche transversale doit s'associer une approche verticale. L'ensemble de la hiérarchie de l'organisme, de la direction jusqu'aux parties intéressées, c'est-à-dire les fournisseurs, partenaires et actionnaires doivent être engagés dans la mise en œuvre du système [4]. Une autre caractéristique des systèmes de management est la formalisation des politiques et procédures de l'organisme afin de pouvoir être audité.

Ces engagements ont un coût en ressources matérielles, humaines et financières. Comment justifier cet investissement ? Les systèmes de management s'appuient sur des guides de bonnes pratiques, mécanismes d'amélioration continue favorisant la capitalisation sur les retours d'expérience, ce qui a pour effet d'accroitre la fiabilité. En outre, l'audit du système de management par un cabinet

d’audit indépendant permet d’établir une relation de confiance entre le client et le fournisseur.

Le fonctionnement du système de management se fait selon le modèle PDCA de l’anglais Plan, Do, Check, Act, en français planifier, faire, contrôler et corriger. Ces quatre phases sont illustrées dans la figure 4 ci-dessous.

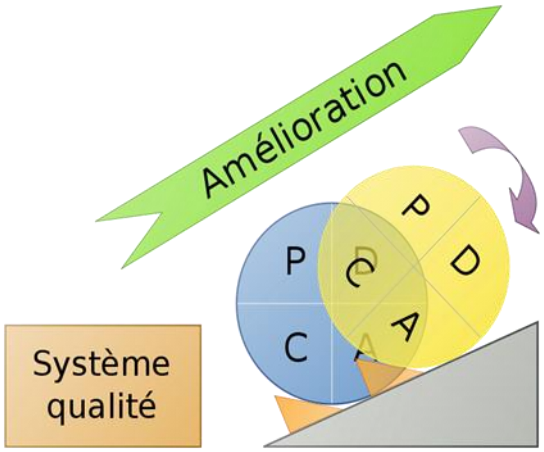


Figure 4 Roue de Deming (PDCA)

- Plan : dire ce que l’on va réaliser dans un domaine particulier.
- Do : faire ce qui a été annoncé.
- Check : vérifier les écarts entre les phases « plan » et «do ».
- Act : ajuster les écarts constatés de la phase « check ».

Une fois que les objectifs fixés par le management sont atteints, il faut s’y tenir dans la durée. La flèche sur la roue Deming, montre qu’un nouveau cycle du processus du système de management doit être entrepris pour y parvenir. Notons que le modèle PDCA s’applique au système de management dans son ensemble ainsi qu’à chacun de ses processus [5].

On retrouve les systèmes de management dans des secteurs d’activités aussi variés que la santé et la sécurité du travail avec la norme OHSAS 18001, l’environnement avec la norme ISO 14001, les services informatiques avec le référentiel ISO/CEI 20000, la sécurité alimentaire avec la norme ISO 22000, la qualité avec la norme ISO 9001 ou encore la sécurité de l’information avec la norme ISO/CEI 27001 que nous allons traiter dans les points suivants.

2.2 Sécurité de l'information

Dans le SMSI, l'information n'est pas restreinte aux systèmes informatiques. L'information est à prendre au sens large du terme. Elle doit être étudiée sous toutes ses formes indépendamment de son support, humain, papier, logiciel, etc. Le terme sécurité doit être compris comme l'ensemble des moyens déployés pour se protéger contre les actes de malveillance. La sécurité du SMSI est définie par la norme ISO 13335-1 à travers les notions de confidentialité, d'intégrité et de disponibilité.

- Confidentialité : seuls les entités, personnes et processus autorisés, ont accès à l'information.
- Intégrité : l'information ne peut être modifiée que par ceux qui en ont le droit.
- Disponibilité : l'information doit être accessible à l'entité, la personne ou le processus qui a un droit d'accès.

Ces trois principes de sécurité peuvent être étendus, les SMSI intègrent d'autres notions telles que l'authentification, la traçabilité, la non-répudiation, l'imputabilité qui constituent des mécanismes de sécurité que l'on déploie en fonction des besoins de sécurité de l'organisme [6].

En conclusion on peut définir les SMSI comme des ensembles d'éléments interactifs permettant à un organisme de fixer une politique et des objectifs de sécurité de l'information, d'appliquer la politique, d'atteindre ces objectifs, de les contrôler et de les améliorer.

Les objectifs sont fixés sur un périmètre défini et doivent être en adéquation avec les besoins de l'organisme concerné, c'est-à-dire que les mesures de sécurité sont à déployer en fonction du contexte, avec un juste dosage, sans exagérations, ni trop de tolérance avec comme finalité la protection des actifs d'information.

Nous avons vu que l'évolution des normes liées à la sécurité de l'information a mené à l'élaboration de SMSI. Dans la partie qui suit nous présentons la famille des normes ISO/CEI 2700x qui font figure de référence en la matière.

3. Les normes de la famille ISO/CEI 2700x

Dans la famille ISO/CEI on trouve deux catégories de normes. Celles qui émettent des exigences : ISO/CEI 27001 et celles qui formulent des recommandations : ISO/CEI 27002. Notons que certaines normes sont encore en cours de rédaction, c'est le cas des normes ISO/CEI 27007 « Audit des SMSI » et ISO/CEI 27008 « Audit des mesures de sécurité ».

Comme représenté sur la figure 5 ci-dessous, la norme ISO/CEI 27001[10] est le centre de gravité des référentiels du SMSI. La norme ISO/CEI 27001 formule les exigences relatives aux SMSI et fournit une liste de mesures de sécurité pouvant être intégrées au système [7].

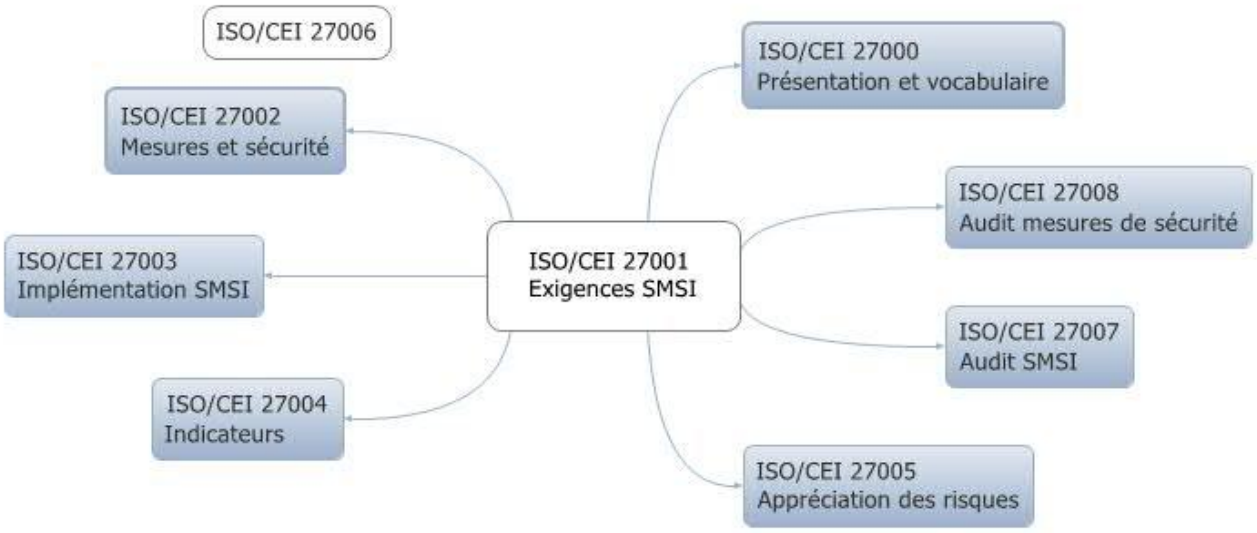


Figure 5 Normes de la famille ISO/CEI 2700x

3.1 L'ISO/CEI 27000

Cette norme a été publiée pour répondre au besoin de définir une terminologie pour les SMSI. Comme nous l'avons vu précédemment, beaucoup de référentiels antérieurs à ISO/CEI 27001 ont été publiés ces dernières années. Ces normes ne fournissent pas de notions, ni de vocabulaire commun, permettant une bonne compréhension des SMSI, c'est ce que propose l'ISO/CEI 27000.

[10] L'ISO/CEI 27001 est développée en détail au chapitre 5.

L'ISO/CEI 27000 est structurée en trois parties. La première, définit 46 termes tels que, la confidentialité, l'intégrité, la disponibilité, l'authenticité, tous principalement axés sur l'appréciation et l'analyse des risques, des menaces, de la vulnérabilité, etc. Par exemple, le mot risque est « la combinaison de la probabilité d'un événement et de ses conséquences ».

La deuxième partie développe la notion de processus avec le modèle PDCA et présente les concepts propres aux SMSI comme par exemple, l'importance de l'engagement de la direction.

La troisième partie, est une présentation de l'ensemble des normes de la famille ISO/CEI 2700x.

3.2 L'ISO/CEI 27002

La norme propose sur onze chapitres, une liste de 133 mesures de sécurité accompagnées chacune de points à aborder pour la mise en place d'un SMSI. Parmi ces chapitres, on a par exemple, la gestion des actifs, la sécurité physique, la sécurité des ressources humaines, la gestion des incidents, la continuité d'activité, la conformité etc. En résumé, l'ISO/CEI 27002 est un guide de bonnes pratiques, une série de préconisations concrètes, abordant les aspects tant organisationnels que techniques, qui permettent de mener à bien les différentes actions dans la mise en place d'un SMSI [8].

3.3 L'ISO/CEI 27003

Publiée en janvier 2010, ISO 27003 facilite la mise en œuvre du SMSI. Elle est utilisée en complément de la norme ISO 27001. L'ISO 27003 propose cinq étapes pour implémenter le SMSI. Ces étapes concernent l'initialisation du projet, sa politique et son périmètre, l'analyse des exigences en matière de sécurité de l'information, l'appréciation des risques et enfin l'élaboration du SMSI. Chacune de ces étapes est divisée en activités qui font l'objet d'une clause contenant :

- un résumé de l'activité (explication de l'étape en question),
- les entrées (tous les documents à utiliser au cours de l'étape),
- les recommandations (détail des points à aborder),
- les sorties (liste des livrables à produire).

En résumé, ISO 27003 propose un grand nombre de points à aborder et énumère les documents à produire et à consulter. Notons que ISO/CEI 27003 reprend les lignes directrices de la norme ISO/CEI 13335 devenue aujourd'hui obsolète.

3.4 L'ISO/CEI 27004

Cette norme concerne la gestion des indicateurs. L'utilisation d'indicateurs dans le domaine de la sécurité est nouvelle. La norme ISO/CEI 27001 impose leur mise en place dans le SMSI mais sans préciser comment et lesquels utiliser. En utilisant les mesures des indicateurs l'objectif est d'identifier les points du SMSI qui nécessitent une amélioration ou une correction.

3.5 L'ISO/CEI 27005

Une des étapes du PDCA les plus difficiles à mettre en œuvre est « l'appréciation des risques[11]». L'ISO/CEI 27001 fixe des objectifs à atteindre pour être en conformité avec la norme, mais ne donne aucune indication sur les moyens d'y parvenir. Nous verrons qu'il existe un nombre important de méthodes d'appréciation des risques qui offrent une démarche formelle et pragmatique. Néanmoins, l'ISO/CEI a souhaité proposer sa propre méthode avec la norme ISO/CEI 27005. L'objectif n'est pas de remplacer ou rendre obsolètes les méthodes existantes mais d'harmoniser le vocabulaire employé. L'ISO/CEI 27005 est constituée de douze chapitres. Les points les plus importants sont traités dans les chapitres sept à douze.

- Chap. 7, établissement du contexte
- Chap. 8, appréciation du risque
- Chap. 9, traitement du risque
- Chap. 10, acceptation du risque
- Chap. 11, communication du risque
- Chap. 12, surveillance et révision du risque

En complément de ces chapitres, viennent s'ajouter six annexes proposant des explications plus détaillées qui présentent des listes de menaces et vulnérabilités

[11] L'appréciation des risques est l'étape 2 de la phase PLAN du PDCA proposé par l'ISO/CEI 27001 étudié au point 5.1..2.

types. L'ISO/CEI 27005 peut aussi servir de référence aux organismes qui cherchent à évaluer une méthode d'appréciation des risques [9].

3.6 L'ISO/CEI 27007

Cette norme est à l'état de brouillon « WD »[12]. On peut cependant avancer qu'elle sera le pendant de la norme générique ISO 19011[13] pour les SMSI. L'ISO/CEI 27007 donnera les lignes directrices pour auditer les SMSI.

3.7 L'ISO/CEI 27008

A l'état de WD, l'ISO/CEI 27008 traitera de l'audit des SMSI en proposant un guide qui permettra de contrôler les mesures de sécurité. Elle fournira pour chacune des mesures de sécurité de la 27002, des moyens d'analyse des besoins en contrôle, en tenant compte de l'importance des risques et des actifs. On pourra ainsi déterminer les mesures à contrôler. Ces points sont importants car les auditeurs internes ou externes doivent contrôler des mesures aussi variées que la gestion des mots de passe, la procédure de gestion des incidents et le suivi de législation en vigueur.

3.8 L'ISO/CEI 27006

Cette norme est une reprise enrichie de la norme ISO 17021[14]. Etant destinée aux cabinets d'audit en charge de certifier les organismes, l'ISO/CEI 27006 est moins connue que l'ISO/CEI 27001 qui s'adresse directement aux organismes souhaitant mettre en place un SMSI. L'ISO/CEI 27006 fournit aux organismes de certification les exigences qu'ils doivent faire respecter pour attribuer à leurs clients une certification. Ces exigences sont par exemple : instaurer des mesures pour garantir la confidentialité des données relatives à leurs clients, établir des

---

[12] WD (working draft) est l'état "projet" de la norme avant sa publication. Voir l'annexe 1 Processus d'élaboration d'une norme ISO suivant Directive ISO/CEI.

[13] ISO 19011 fournit des conseils sur les principes de l'audit, le management des programmes d'audit, la réalisation d'audits de systèmes de management. C'est une norme générique qui peut s'appliquer à différents types de systèmes de management.

[14] ISO 17021 spécifie les principes et les exigences relatives à la compétence, à la cohérence et à l'impartialité lors des audits et lors de la certification de systèmes de management de tous types, ISO 27001 pour les SMSI, 9001 pour le management ou 14001 pour l'environnement.

procédures appropriées pour certifier le SMSI ou encore vérifier régulièrement que les auditeurs soient compétents en matière de sécurité de l'information. Les exigences ne décrivent cependant pas comment l'organisme peut parvenir à la mise en place d'un SMSI. L'organisme doit pour cela procéder à un certain nombre de tâches telles que réaliser un audit des risques, trouver des indicateurs, mettre en œuvre les mesures de sécurité etc. L'organisme a par conséquent besoin d'un guide de bonnes pratiques pour mener à bien son SMSI, et c'est précisément le rôle des normes ISO/CEI 27002 à ISO/CEI 27008.

3.9 Normes ISO/CEI 270xx en préparation

Le tableau 1 ci-dessous donne un aperçu des domaines qui seront couverts par les normes de la famille ISO/CEI 270xx.

Tableau 1 Normes ISO/CEI 270xx en préparation

PROJET	SECTEUR D'ACTIVITE
ISO/CEI 27010	Gestion de la communication inter secteur
ISO/CEI 27031	Continuité d'activité
ISO/CEI 27032	Cyber sécurité
ISO/CEI 27033	Sécurité réseau
ISO/CEI 27034	Sécurité des applications
ISO/CEI 27035	Gestion des incidents
ISO/CEI 27036	Audit des mesures de sécurité du SMSI
ISO/CEI 27037	Gestion des preuves numériques

Il est à noter que certains secteurs comme les télécommunications ou le domaine médical ont développé des normes plus spécifiques à leur métier, ISO/CEI 27011 et ISO/CEI 27799.

PARTIE II. METHODOLOGIE – ISO/CEI 27001

Pour être en conformité avec la norme ISO/CEI 27001, les SMSI doivent répondre à toutes les exigences comprises entre les chapitres 4 et 8 illustrés dans la figure 6 suivante [10].

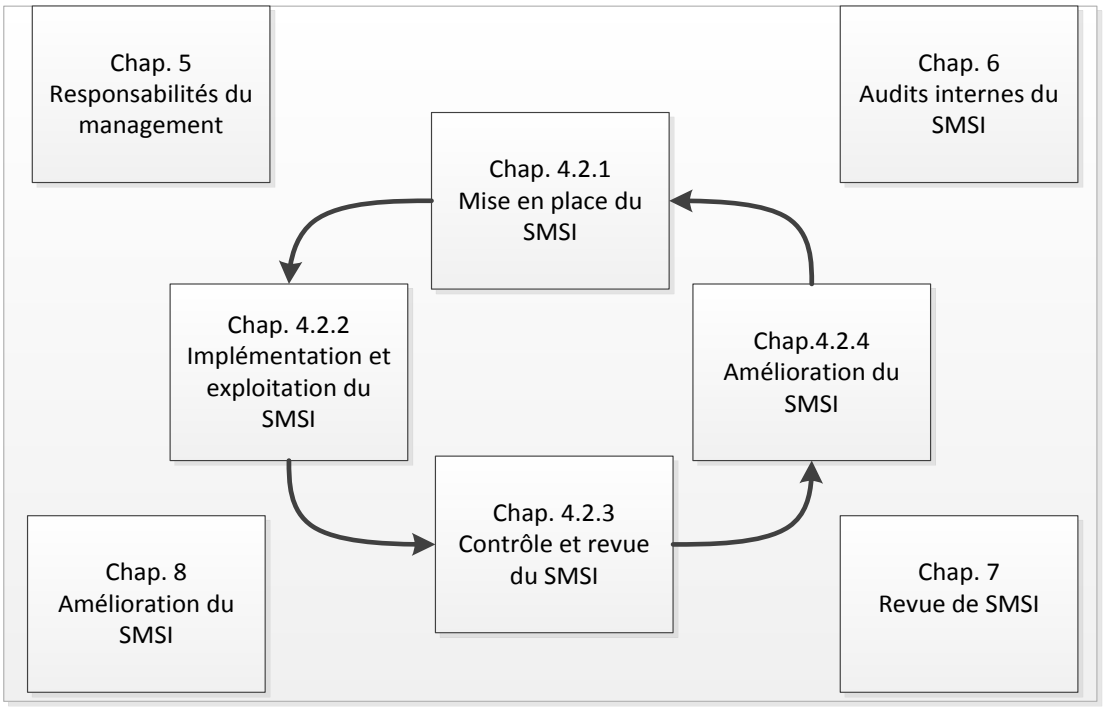


Figure 6 Structure de l'ISO/CEI 27001

- Le chapitre 5 concerne tout ce qui possède un rapport avec les responsabilités du management.
- Le chapitre 6 précise tout ce qui touche aux audits internes.
- Le chapitre 7 développe les aspects relatifs aux revues des différents éléments du SMSI.
- Le chapitre 8 définit les notions d'actions correctives et préventives.
- Le chapitre 4 est au centre de la norme, il regroupe les quatre phases du PDCA (PLAN, DO, CHECK, ACT) de la roue de Deming.

1. Phase « PLAN » du PDCA

La phase « Plan » du PDCA consiste à fixer les objectifs du SMSI en suivant quatre grandes étapes, la politique et le périmètre du SMSI, l'appréciation des risques, le traitement des risques décidé en tenant en compte des risques résiduels et la sélection des mesures de sécurité présentées dans le SoA[15] (Statement of Applicability).

Dans la figure 7 ci-dessous, une vue du déroulement de la phase Plan.

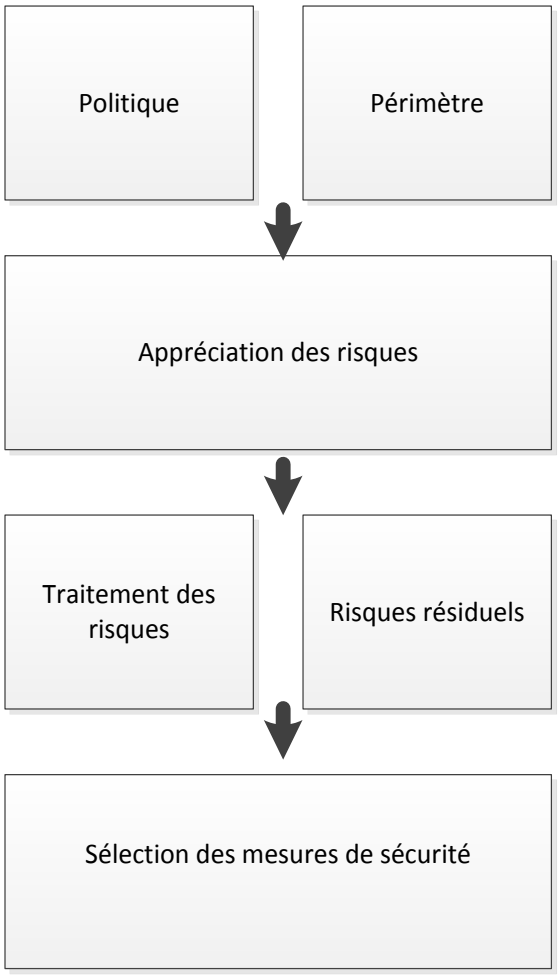


Figure 7 Etapes de la phase Plan du PDCA

[15] SoA (Statement of Applicability) est un document sous forme de tableau qui énumère les mesures de sécurité du SMSI ainsi que celles non appliquées.

1.1 Politique et périmètre du SMSI

La première étape consiste à définir la politique et le périmètre du SMSI.

La politique est là pour préciser le niveau de sécurité qui sera appliqué au sein du périmètre du SMSI. La norme ne fixe pas d'exigences sur le périmètre, il peut être restreint ou couvrir l'ensemble des activités de l'organisme. L'objectif est d'y inclure les activités pour lesquelles les parties prenantes exigent un certain niveau de confiance.

1.2 Appréciation des risques

La deuxième étape concerne un des points les plus importants de l'ISO/CEI 27001, l'appréciation des risques. Le problème de l'appréciation des risques n'est pas nouveau et est traité par de nombreuses méthodes développées dans différents secteurs privés, académiques et agences gouvernementales. Certaines méthodes sont très répandues dans les organismes. En France, les plus connues sont EBIOS et MEHARI, aux Etats-Unis, OCTAVE. L'ISO/CEI propose aussi une méthode, la norme ISO/CEI 27005, mais ne l'impose pas. L'ISO/CEI 27001 ne fait que fixer un cahier des charges spécifiant chacune des étapes clefs de l'appréciation des risques. L'organisme a le libre choix, de développer sa propre méthode en suivant les objectifs fixés par l'ISO/CEI 27001 ou d'en appliquer une déjà éprouvée.

Dans les points suivants nous détaillons le processus d'appréciation des risques avant de donner trois exemples de méthodes parmi les plus connues.

1.2.1 Processus d'appréciation des risques

Le processus d'appréciation des risques se déroule en sept étapes, illustrées dans figure 8 ci-dessous.

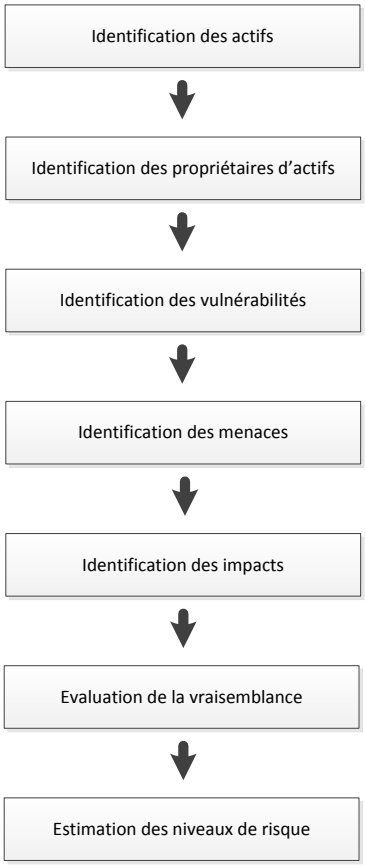


Figure 8 Processus d'appréciation des risques

La première étape consiste dresser une liste de tous les actifs qui ont une importance en matière d'information au sein du SMSI. On distingue généralement six catégories d'actifs.

- Matériel, pour tous les équipements réseau et système.
- Physique, pour les bureaux, lieux de production, de livraisons.
- Logiciel, pour les bases de données, fichiers, les systèmes d'exploitation.
- Humain, pour tous les collaborateurs de l'organisme.
- Documents, pour les documents papier, manuels d'utilisation.
- Immatériel, pour le savoir faire de l'organisme.

La deuxième étape vise à attribuer pour chaque actif d'information un « propriétaire ». La norme définit le propriétaire comme étant la personne qui connaît le mieux la valeur et les conséquences d'une compromission en termes de disponibilité, d'intégrité et de confidentialité de l'actif.

La troisième étape est l'identification des vulnérabilités des actifs recensés. La vulnérabilité est la propriété intrinsèque du bien qui l'expose aux menaces. A titre d'exemple, un ordinateur portable est vulnérable au vol mais sa vulnérabilité n'est pas le vol mais sa portabilité. Dans ce cas l'identification de la vulnérabilité est la portabilité.

La quatrième étape est l'identification des menaces qui pèsent sur les actifs d'information précédemment recensés. Si l'on reprend l'exemple de l'ordinateur portable, la menace est dans ce cas le vol.

La cinquième étape vise à évaluer l'impact d'une perte de la confidentialité, de la disponibilité ou de l'intégrité sur les actifs. Pour mesurer cet impact on peut par exemple utiliser une matrice des risques[16], la norme n'impose aucun critère de mesure.

La sixième étape demande d'évaluer la vraisemblance des précédentes étapes du processus en plaçant dans leur contexte les actifs. Il s'agit par exemple de considérer les mesures de sécurité déjà en vigueur dans l'organisme. Si l'ordinateur portable possède une clef d'authentification, un cryptage de ses données ou un accès VPN[17] pour travailler, alors la vraisemblance d'observer un impact sur la confidentialité, la disponibilité ou l'intégrité de ses données est limitée.

La septième étape consiste à attribuer une note finale reflétant les risques pour chacun des actifs d'information. La norme n'impose aucune formule, on peut par

[16] La matrice des risques permet de mettre en rapport les niveaux de risques définis par la direction avec ceux identifiés dans l'appréciation des risques.

[17] VPN (réseau privé virtuel) est vu comme une extension des réseaux locaux et préserve la sécurité logique que l'on peut avoir à l'intérieur d'un réseau local.

exemple utiliser un code couleur (rouge pour un niveau de risque très élevé, orange pour moyen et vert pour faible [9].

Dans le point suivant, nous présentons trois méthodes connues et largement employées par les organismes pour l'appréciation des risques de leur SMSI.

1.2.2 Méthodes d'appréciation des risques

En 2004, une étude du CLUSIF (Club de la Sécurité de l'Information Français) dénombrait plus de deux cents méthodes d'appréciation des risques. Nous en avons retenu trois, EBIOS, MEHARI et OCTAVE qui pour l'ENISA (European Network and Information Security Agency) figurent parmi les plus utilisées [11].

EBIOS

Développée dans les années 90 sous l'autorité de l'agence française ANSSI (Agence nationale de la sécurité des systèmes d'information), cette méthode est l' « Expression des Besoins et Identification des Objectifs de Sécurité ». Elle permet d'apprécier, de traiter et communiquer sur les risques au sein d'un SMSI. L'ANSSI et le Club EBIOS[18] proposent en libre accès sur leur site web toute la documentation[19] ainsi qu'un logiciel libre[20] facilitant l'utilisation de la méthode. L'approche de la méthode est itérative, chaque module peut être révisé, amélioré et tenu à jour de manière continue [12].

[18] Le club EBIOS sur le site : www.club-ebios.org, est une communauté francophone de la gestion des risques basée sur la méthode EBIOS.

[19] La méthode EBIOS est publiée sous la forme d'un guide et de bases de connaissances riches et adaptables (types de biens, d'impacts, de sources de menaces, de menaces, de vulnérabilités et de mesures de sécurité).

[20] Logiciel libre EBIOS, intègre une base de connaissances permettant de saisir les hypothèses et de synthétiser les résultats.

EBIOS se compose de cinq modules représentés dans la figure 9 ci-dessous :

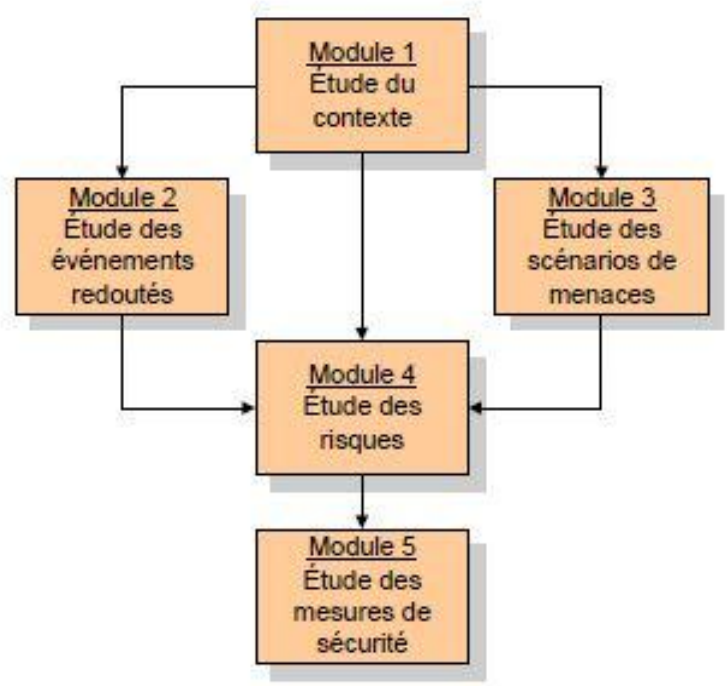


Figure 9 Modules d'EBIOS

Module 1 : il concerne l'étude du contexte. Il s'agit de détailler l'organisation, les missions, les contraintes et les métiers pour rendre applicable et cohérent le choix des objectifs de sécurité. Le point suivant consiste à identifier les fonctions estimées sensibles, la perte, le dysfonctionnement ou la divulgation d'informations qui peuvent avoir des répercussions sur le bon fonctionnement de l'organisme. Enfin, on répertorie sous forme de matrice les entités[21] techniques propres au SMSI (matériel, logiciels, réseaux) ainsi que les entités organisationnelles (groupes de collaborateurs) pour établir les liens entre les éléments essentiels et les entités.

Module 2 : il concerne l'étude des événements redoutés. Cette étape permet de définir les besoins de sécurité des éléments essentiels précédemment identifiés. On quantifie les besoins sur une échelle de 0 à 4 à l'aide d'un questionnaire que l'on adresse aux collaborateurs de l'organisme. Les besoins sont sélectionnés sur des critères de sécurité tels que la disponibilité, l'intégrité, la confidentialité et la

[21] Terme employé par la méthode pour décrire un constituant de l'organisme.

non-répudiation ainsi que sur des critères d'impacts[22] (interruption de services, dommages matériels).

Module 3 : consiste à étudier les scénarios de menaces. Estimer, évaluer les menaces (incendie, perte d'alimentation électrique, divulgation d'information etc.) et identifier les objectifs de sécurité qu'il faut atteindre pour les traiter. EBIOS fournit une liste de menaces que l'on associe aux éléments essentiels définis dans le module 1. Puis on attribue à chaque élément un niveau de vulnérabilité sur une échelle de 0 à 4.

Module 4 : il vise à étudier les risques. Cette étape permet de dresser une cartographie des risques. Elle explique aussi comment traiter le risque. Estimer, évaluer les risques puis identifier les objectifs de sécurité à atteindre pour les traiter.

Module 5 : il concerne l'étude des mesures de sécurité. Cette dernière étape explique comment appliquer les mesures de sécurité à mettre en œuvre, comment planifier la mise en œuvre de ces mesures et comment valider le traitement des risques résiduels.

En conclusion, la méthode EBIOS par son caractère exhaustif, permet de formaliser tout le SMSI et son environnement. Cette méthode contribue à formuler une politique de sécurité du système d'information. C'est une des méthodes pour mettre en œuvre le cadre défini par l'ISO/CEI 27005. Elle répond aux exigences de l'ISO/CEI 27001 et peut exploiter les mesures de sécurité de l'ISO/CEI 27002.

MEHARI

La méthode MEHARI (Méthode Harmonisée d'Analyse de Risques) a été développée dans les années 1990 par le CLUSIF[23] (Club de la Sécurité de l'Information Français). A l'origine, cette méthode ne traitait que de l'analyse des

[22] Critères d'impacts, une liste est fournit par EBIOS.

[23] Club de la Sécurité de l'Information Français, est une association française d'entreprises et de collectivités réunie en groupes de réflexion et d'échanges autour de différents domaines de la sécurité de l'information : gestion des risques, politiques de sécurité, cybercriminalité, intelligence économique.

risques. Elle a évolué pour permettre une gestion de la sécurité de l'organisme dans un environnement ouvert et géographiquement réparti.

MEHARI a été adoptée par des milliers d'organismes à travers le monde et reste la méthode la plus utilisée en France, en particulier dans l'industrie. L'utilisation et la distribution de son logiciel sont libres. En outre, certaines bases de connaissances sont disponibles et une étude illustre la méthode pour faciliter son utilisation.

MEHARI s'appuie sur deux méthodes anciennes aujourd'hui abandonnées, appelées MARION[24] (Méthode d'Analyse de Risques Informatiques Optimisée par Niveau) et MELISA (Méthode d'évaluation de la Vulnérabilité Résiduelle des Systèmes d'armement).

Contrairement à la méthode EBIOS, MEHARI repose sur des scénarios de risques qui permettent d'identifier les risques potentiels au sein de l'organisme. Elle est définie comme une boîte à outils conçue pour la gestion de la sécurité. En fonction des besoins, des choix d'orientation, de politique de l'organisation ou simplement des circonstances, la méthode veille à ce qu'une solution d'appréciation des risques appropriée puisse être élaborée. La méthode est présentée sous la forme d'un ensemble que l'on appelle modules, centrés sur l'évaluation des risques et leur gestion [13].

[24] Marion, méthode d'audit, proposée depuis 1983 par le CLUSIF, visant à évaluer le niveau de sécurité informatique d'une entreprise.

Chaque module peut être utilisé indépendamment ou en combinaison et conduit généralement à un plan d'action, comme le montre la figure 10 ci-dessous.

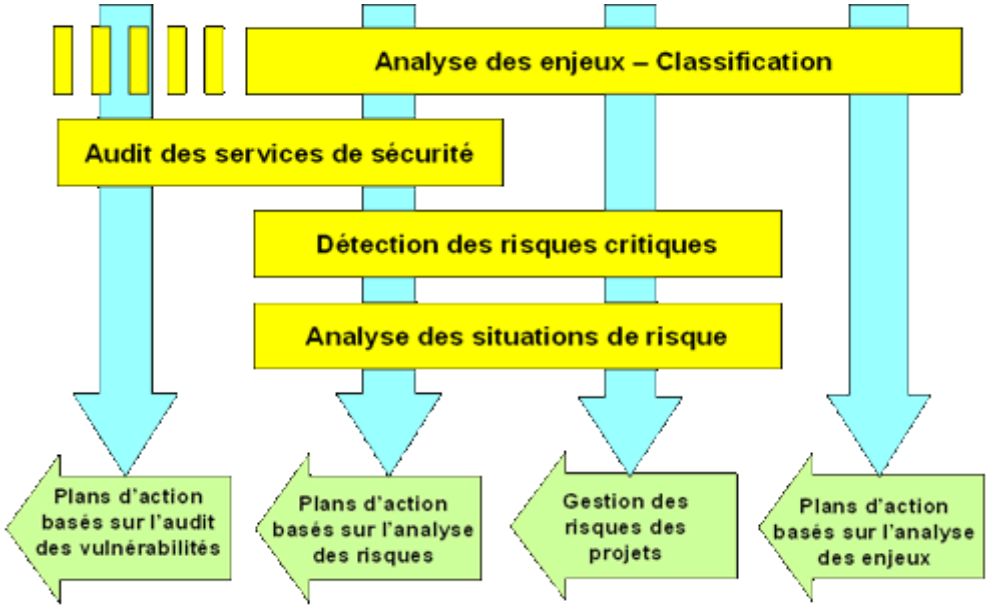


Figure 10 Utilisation des modules de MEHARI

Les modules sont les suivants :

Le module « analyse des enjeux et classification » permet d'identifier et d'analyser les ressources[25] de l'organisation ainsi que les enjeux concernant la sécurité. Ce module s'appuie sur quatre principes.

- Le premier, consiste à présenter les métriques de la méthode (quantification de la potentialité et de l'impact, appréciation des niveaux de risques).
- Le deuxième, vise à dresser une liste des priorités à respecter. Ces objectifs de sécurité, établis par la direction, dépendent des spécificités de l'organisme.

[25] Ressources peuvent être aussi bien les locaux, qu'un serveur ou un groupe d'employés, la norme ISO/CEI 27001 utilise le terme « actif » pour définir les ressources.

- Le troisième, concerne la classification des ressources de l'entreprise. Les ressources sont identifiées et classées en fonction de leur impact en termes de disponibilité, d'intégrité et de confidentialité.

- Le quatrième principe est la mise en œuvre de la charte et de la politique de sécurité de l'organisme.

Le second module concerne les services de sécurité. A l'aide de la base de connaissance des mesures de sécurité de la méthode on corrige les points faibles par des plans d'action, on évalue l'efficacité des mesures mises en œuvre, on prépare l'analyse des risques induits par les faiblesses mises en évidence, et on termine par un audit des vulnérabilités sur la base des normes en usage.

Le troisième module est l'analyse des situations de risque qui permet de faire une synthèse des actions de sécurité préalablement menées. On sélectionne des indicateurs en fonction des objectifs fixés dans le premier module. Ces indicateurs permettent de comparer les résultats obtenus aux objectifs fixés. Cette dernière étape offre un suivi dans le temps du niveau de sécurité de l'information.

En conclusion, la méthode MEHARI, se caractérise par une démarche « descendante », c'est-à-dire une délégation des décisions de la direction vers les entités opérationnelles. Elle convient bien aux organisations multi-environnements. Cette méthode est conforme au cahier des charges imposé par l'ISO/CEI 27001.

OCTAVE

OCTAVE (Operationally Critical Threat, Asset, and Vulnerability Evaluation) est une méthode d'évaluation du risque développée en 1999 par le Software Engineering Institute de l'Université Carnegie Mellon aux Etats-Unis. La documentation est publique et la dernière version 2.0 date de 2001. Par rapport aux autres méthodes, sa dernière mise à jour est ancienne mais demeure pertinente. Cette méthode se caractérise par une analyse des risques qui peut être réalisée exclusivement à partir des ressources internes. Elle permet, comme les autres méthodes, l'évaluation des menaces et les vulnérabilités des actifs

d'information recensés [14]. La méthode est développée autour de trois phases comme le montre la figure 11 ci-dessous.

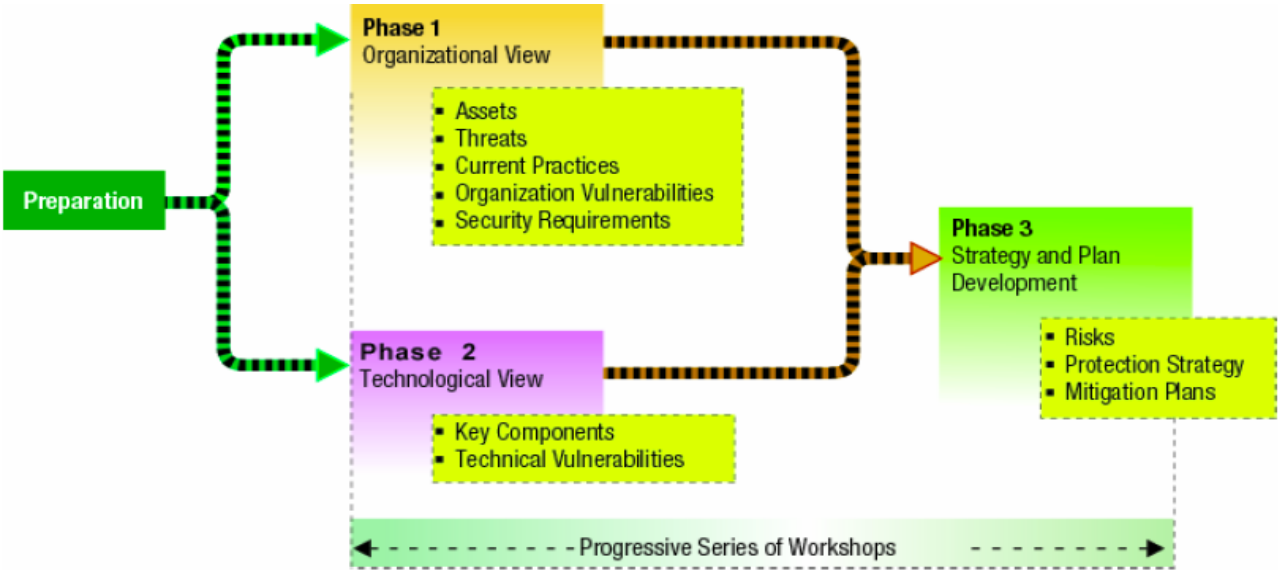


Figure 11 Phases de la méthode OCTAVE

La première phase a pour objectif d'identifier les actifs les plus importants en fonction des menaces et vulnérabilités qui leurs sont associés. On interroge les collaborateurs à tous les niveaux de l'organisme pour rassembler le plus d'informations possibles. Cela permet de dresser des profils de menaces sur les actifs et les besoins en sécurité sur la base de critères tels que la disponibilité, l'intégrité et de confidentialité de l'information.

La deuxième phase a pour but l'identification des vulnérabilités dites techniques. On évalue par exemple, l'infrastructure réseau, on examine les chemins d'accès de chaque actif critique en vue d'obtenir une mesure de la vulnérabilité des infrastructures.

La troisième phase de la méthode décline le développement de la stratégie de la sécurité et sa planification par une analyse de risque et la mise en place des mesures de sécurité.

En conclusion, la méthode OCTAVE offre une « stratégie » de protection pour l'organisationnel et l'opérationnel. Un plan de gestion des risques pour les infrastructures liées à l'information et un plan d'action pour l'ensemble de

l'organisation. La méthode vise les administrations et les grands comptes. Il existe aussi une version adaptée pour les PME.

En résumé, l'approche systématique et structurée de ces méthodes, permet d'évaluer les vulnérabilités d'un système d'information, de quantifier les risques et d'aider à la mise en œuvre d'un processus d'appréciation des risques du SMSI. Cependant, si l'objectif est commun, les termes et expressions employées varient d'une méthode à l'autre. L'ISO/CEI 27001 et 27005 apporte une cohérence à l'ensemble du processus ce qui facilite sa compréhension.

Après avoir étudié en détail l'étape 2 de la phase « Plan » du PDCA, nous allons poursuivre avec l'étape 3 qui concerne le traitement des risques et l'identification des risques résiduels.

1.3 Traitement des risques

La troisième étape concerne le choix du traitement des risques. L'ISO/CEI 27001 a identifié quatre traitements possibles du risque, l'acceptation, l'évitement, le transfert et la réduction.

- « Accepter » le risque revient à ne déployer aucune mesure de sécurité autre que celles déjà en place. Cette décision peut être justifiée si le vol de données dans un cas précis n'a pas d'impact sur l'organisme.

- « Eviter » le risque consiste à supprimer par exemple l'activité ou le matériel offrant un risque.

- « Transférer » un risque par souscription d'une assurance ou par sous-traitance. Ces moyens de transfert du risque sont souvent employés quand l'organisme ne peut ou ne souhaite pas mettre en place les mesures de sécurité qui permettraient de le réduire.

- « Réduire » le risque consiste à prendre des mesures techniques et organisationnelles pour ramener à un niveau acceptable le risque. C'est le traitement le plus courant.

Il existe d'autres traitements du risque possibles mais pour être en conformité avec la norme, il faut en priorité considérer ceux que nous venons de citer.

Après avoir sélectionné le traitement et mis en place les mesures de sécurité, un risque peut persister. Il convient de traiter ce risque comme les autres c'est-à-dire, l'accepter, l'éviter, le transférer ou le réduire.

1.4 Sélection des mesures de sécurité

L'étape 4 est la dernière étape de la phase « Plan » du PDCA, elle consiste à sélectionner les mesures de sécurité. La norme ISO/CEI 27001 propose dans son annexe A, 133 mesures de sécurité réparties sur onze chapitres. A ce stade, le travail consiste à dresser un tableau, appelé SoA (Statement of Applicability) dans lequel figurent les 133 mesures qu'il faut déclarer applicables ou non applicables, pour réduire les risques du SMSI. Notons que les 133 mesures proposées par l'ISO/CEI 27001 répertorient presque tout ce qui peut être entrepris en matière de sécurité de l'information cependant, cette liste ne comporte pas d'exemples ni d'explications sur le déploiement des mesures à entreprendre. L'ISO/CEI 27002 répond en partie à ce besoin en fournissant une série de préconisations et d'exemples techniques et organisationnels qui couvrent la liste de l'ISO/CEI 27001.

Une fois choisie la politique et le périmètre du SMSI, appréciés et traités les risques, et sélectionnées les 133 mesures de sécurité dans le tableau SoA, il faut mettre en œuvre les objectifs fixés de la phase « Plan » du PDCA.

Il s'agit de la phase « Do » du PDCA.

2. Phase « DO » du PDCA

Cette phase consiste à décrire la mise en œuvre des mesures de sécurité sélectionnées dans le SoA à travers quatre étapes.

2.1 Plan de traitement

Il faut premièrement gérer l'interdépendance des actions à entreprendre. Certaines mesures sont partiellement ou déjà en place, d'autres doivent être intégralement déployées ou nécessitent la mise en œuvre d'une autre action avant de pouvoir être lancées. Ce travail revient à établir un plan de traitement qui peut

être assimilé à de la gestion de projet. Une fois ce travail effectué, il faut déployer les mesures de sécurité en suivant le plan de traitement.

Par la suite, le responsable de projet doit définir des « mesures d'efficacité » pour contrôler le bon fonctionnement du SMSI.

2.2 Choix des indicateurs

Ce point consiste à mettre en place des indicateurs de performance pour vérifier l'efficacité des mesures de sécurité ainsi que des indicateurs de conformité pour contrôler la conformité du SMSI. Trouver de bons indicateurs n'est pas une tâche facile. La norme ne préconise pas d'indicateurs précis à utiliser mais l'ISO/CEI 27004 propose une démarche[26] qui peut aider à les sélectionner [10].

L'étape suivante concerne la sensibilisation des collaborateurs aux principes de la sécurité de l'information.

2.3 Formation et sensibilisation des collaborateurs

Nous avons vu que les mesures de sécurité couvrent de nombreux domaines allant de la sécurité organisationnelle à la sécurité physique, en passant par la sécurité des systèmes réseaux etc. Les collaborateurs doivent maîtriser les outils de sécurité déployés dans les domaines très variés. Une formation du personnel peut s'avérer nécessaire.

La sensibilisation à la sécurité du système d'information concerne tous les collaborateurs. Elle peut débuter par un rappel des engagements de leur entreprise en matière de sécurité et se poursuivre par une liste de conseils tels que le respect de certaines règles de sécurité pour les mots de passe et l'environnement de travail.

2.4 Maintenance du SMSI

La maintenance consiste à garantir le bon fonctionnement de chacun des processus du SMSI et vérifier que leur documentation est à jour. Cela permet à l'auditeur externe de contrôler la gestion du SMSI. Il est à noter que tous les systèmes de management ISO sont concernés par la maintenance [7].

---

[26] La démarche de l'ISO/CEI 27004 est traitée au point 3.4 de la partie I.

A ce stade de l'avancement du SMSI, les mesures identifiées du SoA fonctionnent, les indicateurs sont implémentés et les collaborateurs de l'organisme formés et sensibilisés à la sécurité du SMSI, nous pouvons poursuivre avec la phase « Check » du PDCA.

3. Phase « CHECK » du PDCA

La phase « Check » du PDCA concerne les moyens de contrôle à mettre en place pour assurer « l'efficacité » du SMSI et sa « conformité » au cahier des charges de la norme ISO/CEI 27001 [10]. Pour répondre à ces deux exigences de la norme, les organismes emploient le contrôle et les audits internes ainsi que les revues de direction.

3.1 Les audits internes

L'audit interne peut s'organiser avec le personnel de l'organisme ou être sous traité à un cabinet conseil. Si l'audit est confié à un collaborateur, il ne faut pas que ce dernier puisse auditer un processus dans lequel il est impliqué au niveau de sa mise en œuvre ou de son exploitation. L'audit a pour but de contrôler la conformité et l'efficacité du SMSI en recherchant les écarts entre la documentation du système (enregistrement, procédures, etc.) et les activités de l'organisme. La norme exige que la méthode utilisée pour l'audit soit documentée dans une procédure et que les rapports soient enregistrés pour être utilisés lors des revues de direction.

3.2 Les contrôles internes

L'objectif du contrôle interne est de s'assurer au quotidien que les collaborateurs appliquent correctement leurs procédures. Contrairement à l'audit interne qui est planifié longtemps à l'avance, les contrôles internes sont inopinés.

3.3 Revues de direction

La revue est une réunion annuelle qui permet aux dirigeants de l'organisme d'analyser les événements qui se sont déroulés sur l'année en cours. Les points passés en revue sont généralement :

- les résultats des audits,

- le retour des parties prenantes,
- l'état des lieux sur les actions préventives et correctives,
- les menaces mal appréhendées lors de l'appréciation des risques,
- l'interprétation des indicateurs et les changements survenus dans l'organisme.

A partir de ces informations la direction peut fixer de nouveaux objectifs et allouer de nouvelles ressources (financières, humaines et matérielles).

Les contrôles de la phase « Check » peuvent faire apparaître des dysfonctionnements du SMSI. Cela peut être un écart entre les exigences de la norme et le système de management ou des mesures de sécurité inefficaces. C'est dans la phase « Act » du PDCA que l'on réduit les dysfonctionnements par des actions correctives, préventives ou d'améliorations.

4. Phase « ACT » du PDCA

4.1 Actions correctives

On intervient de manière « corrective » lorsqu'un dysfonctionnement ou un écart est constaté. On agit premièrement sur les effets pour corriger cet écart ou dysfonctionnement, puis sur les causes pour éviter qu'ils ne se répètent.

4.2 Actions préventives

On emploie les actions préventives quand une situation à risque est détectée. On agit sur les causes avant que l'écart ou le dysfonctionnement ne se produisent.

4.3 Actions d'améliorations

Les actions d'améliorations ont pour objectif l'amélioration de la performance du SMSI.

Les résultats des différentes actions doivent être enregistrés et communiqués aux parties prenantes. Ces actions contribuent à rendre plus efficace et performant le SMSI.

Nous avons présenté la méthode et les exigences de la norme ISO/CEI 27001 pour mettre en œuvre un SMSI. Dans la partie qui suit, nous allons voir comment le SMSI a été déployé dans une PME hautement spécialisée dans le domaine de la micro optique, SMO (Suss MicroOptics).

PARTIE III. MISE EN ŒUVRE DU SMSI DE SMO

1. Contexte

Il y a quatre ans, SMO (Suss MicroOptics) a mis en œuvre un système de management de la Qualité qui lui a permis d'être certifiée ISO 9001 en 2007. Cette démarche « Qualité » a engendré un travail de réflexion sur le management des risques et de la sécurité liés à l'information ce qui a permis dans un premier temps d'inclure au système Qualité ISO 9001 quelques procédures de sécurité. Par la suite, cette démarche a été encouragée par certains clients et fournisseurs de SMO qui avaient mis ou mettaient en œuvre un SMSI. C'est dans ce contexte que la direction a décidé en 2009 de lancer une étude pour établir une liste des actifs de l'entreprise liés à l'information afin d'évaluer les menaces, leur niveau et probabilité d'impact et les solutions à mettre en œuvre pour limiter ces risques.

2. Etude préalable

Le projet de mise en œuvre d'un SMSI est complexe, comme nous avons pu le voir à travers l'étude de la norme ISO/CEI 27001. Le projet couvre de nombreux domaines, matériel, physique, logiciel, humain, documentaire et immatériel. Aussi, avant d'aborder la première étape qui consiste à établir la politique de sécurité et le périmètre du SMSI, nous avons effectué, à partir d'entretiens avec les principaux responsables métiers, un diagnostic de l'existant et une analyse des écarts entre les pratiques de SMO en matière de sécurité de l'information, et les mesures de sécurité de la norme ISO/CEI 27001. Le résultat de ce travail est documenté dans le système qualité SMO *I 8.2.005 Analyse des risques V1.1*. et le SoM[27] (Statement of Maturity). C'est sur la base de ce document qu'ont pu être établis la politique et le périmètre du SMSI de SMO.

Dans la partie qui suit nous donnons un aperçu du diagnostic de SMO à travers la présentation de son activité, son marché, sa clientèle, sa structure organisationnelle et informatique.

2.1 SMO (Suss MicroOptics)

Créé en 1999, SMO est un des leaders mondiaux dans le domaine de la micro-optique. Elle est une filiale de SMO fournisseur d'équipements pour l'industrie des

[27] Statement of Maturity est présenté au chapitre 5.1, partie III.

semi-conducteurs et un spin-off de l'IMT[28] (Institut de Microtechnique de Neuchâtel) en Suisse.

2.2 Activité

L'activité cible de SMO est la conception, le design et la fabrication de matrices de microlentilles réfractives et systèmes d'illuminations standards ou customisés[29]. Aujourd'hui, les matrices de microlentilles sont des composantes essentielles des systèmes optiques. Elles sont omniprésentes, dans les systèmes de télécommunications ou les machines de vision telles que les caméras numériques.

Les matrices de microlentilles sont par exemple souvent utilisées pour accroître l'efficacité de la réception de la lumière sur les capteurs CCD[30].

2.3 Marché

Le marché cible de SMO est l'industrie des lasers, les semi-conducteurs, le couplage de fibres, les Shack Hartmann[31], la métrologie, la microscopie confocale, les lampes UVC, les senseurs chimiques, les MOEMS (Microsystèmes Opto-Electro-Mécaniques), l'imagerie (micro-cameras).

---

[28] L'IMT est un institut technique et scientifique dépendant de l'École polytechnique fédérale de Lausanne.

[29] Répondant aux spécifications techniques du client.

[30] Le CCD Charge-Coupled Device, transforme les photons lumineux qu'il reçoit en paires électron-trou par effet photoélectrique dans le substrat semi-conducteur, puis collecte les électrons dans le puits de potentiel maintenu à chaque photosite. Le nombre d'électrons collectés est proportionnel à la quantité de lumière reçue.

[31] Un Shack-Hartmann est un instrument permettant de mesurer la déformation du front d'onde du faisceau optique.

2.4 Structure organisationnelle

La structure organisationnelle de la société se présente de la façon suivante figure 12.

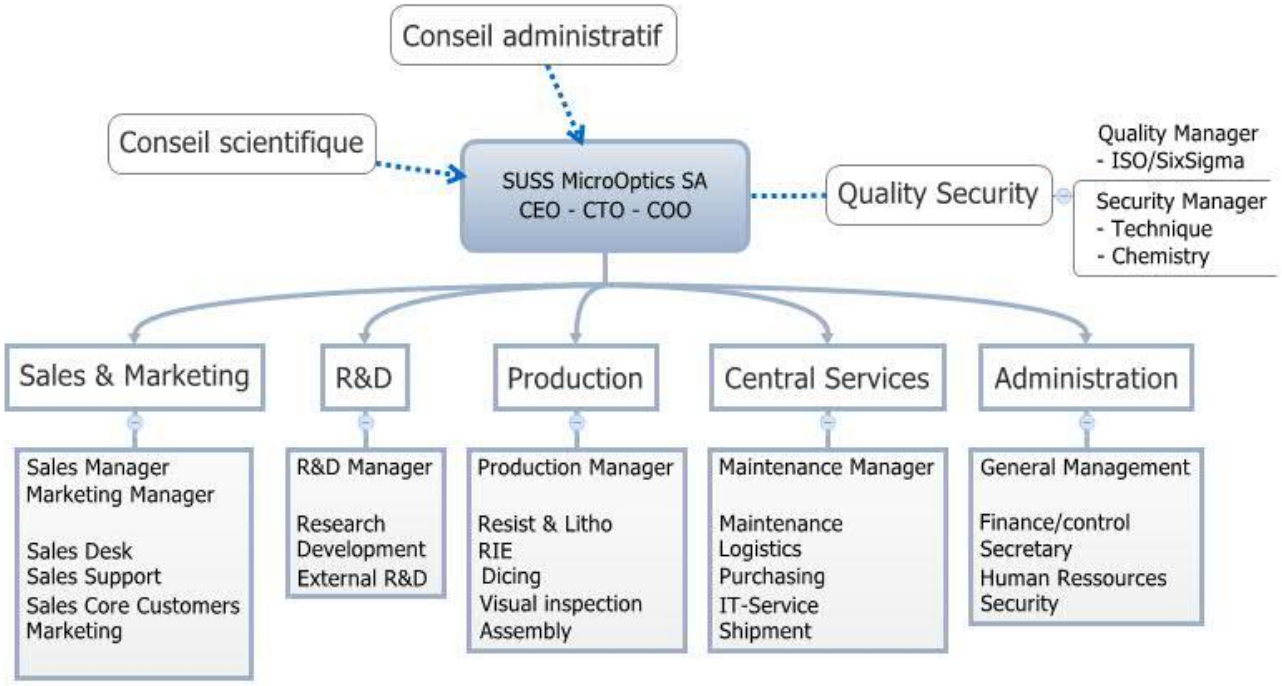


Figure 12 Organigramme fonctionnel SMO

Un conseil scientifique et un conseil administratif assistent les trois directeurs de SMO. Le directeur (CEO) de SMO dirige la société tout en collaborant aux travaux de recherche et développement, de vente et marketing, aux services généraux et à l'administration. Il est assisté par deux adjoints, un directeur technique (CTO) et opérationnel (COO), respectivement physicien et ingénieur en sciences optiques.

Le département « Ventes et Marketing » est constituée d'une équipe de quatre ingénieurs des filières optique et électronique, animée par un ingénieur en optoélectronique et une économiste qui s'occupe aussi en partie de la comptabilité et de la certification ISO.

Le département « Recherche & Développement » regroupe cinq ingénieurs des filières optique, électronique et physique. L'activité s'organise autour de l'homogénéisation des faisceaux laser, des systèmes d'illumination, des tests des

éléments optiques ainsi que la recherche de nouveaux composants, de nouvelles technologies, et systèmes de test.

Le département production est constitué de sept ingénieurs, trois techniciens et trois opérateurs travaillant essentiellement en « salle blanche[32] ». Leur travail s'articule autour du développement, de la qualification, de l'optimisation et de la mise en production de processus « photoresist[33] – photolithographie[34] » pour fabrication des éléments micro-optiques.

Le département « Central services » services généraux, possède trois techniciens, une économiste et quatre ingénieurs. Le département s'occupe de la mise en service, de l'entretien, de la maintenance et de la réparation des machines de production en salle blanche avec la mise en place et le suivi de la documentation correspondante. Les services généraux gèrent aussi le stock des pièces détachées et des consommables du parc machines. Un technicien s'occupe de la coordination avec les services techniques du CSEM[35] pour les questions touchant aux installations mise à disposition par CSEM (air comprimée, N2, vide, circuit de refroidissement, extraction, etc...). Un autre technicien est responsable de la maintenance, l'entretien et l'organisation des ordinateurs, du réseau informatique, des logiciels, serveurs et PC.

---

[32] Salle blanche est une pièce où la concentration particulaire est maîtrisée afin de minimiser l'introduction, la génération, la rétention de particules à l'intérieur.

[33] Photorésist ou photorésine, est un matériau photosensible utilisé dans de nombreux procédés industriels, comme la photolithographie ou la photogravure afin de former un revêtement protecteur ajouré à la surface d'un substrat.

[34] La photolithographie est l'ensemble des opérations permettant de délimiter l'extension latérale des matériaux sur la surface d'un substrat semi-conducteur, dont la structure est plus ou moins bidimensionnelle car basée sur l'empilement de couches à la surface d'une plaquette de silicium.

[35] CSEM, Centre Suisse d 'Electronique et de Microtechnique spécialisé dans les nanotechnologies, la microélectronique, l'ingénierie système et les technologies de communication. Il offre à ses partenaires de l'industrie des solutions et des services issus de la recherche appliquée.

Le département de l'administration est dirigé par le CEO et son adjoint assisté du secrétariat et de l'économiste pour la gestion des ressources humaines, des finances et système Qualité. En outre, un responsable chimie et un responsable machines, tous deux ingénieurs, gèrent la sécurité des postes de travail et assurent la formation et la maintenance de la documentation qualité correspondante.

2.5 La clientèle

SMO a plus d'une centaine de clients en Asie, Europe et aux Etats-Unis. L'entreprise distingue trois types de clients sur la base de leurs besoins et attentes.

- Les clients dont les produits nécessitent une ingénierie de haute précision avec une prise en charge de la conception optique ainsi que la R&D pour faire évoluer leurs produits.
- Les clients avec un produit nécessitant une ingénierie plus simple.
- Les clients qui achètent des produits standard avec un délai de livraison court et qui n'établiront pas nécessairement une collaboration en R&D avec SMO.

2.6 La structure informatique

Le réseau LAN (Local Area Network) couvre tous les locaux de SMO. Il est composé de cinq imprimantes, trente six postes de travail, deux pare-feu, deux modems et quatre serveurs. Trois serveurs sont situés dans un local qui leur est dédié et le quatrième est dans le bureau du responsable TI.

La figure 13 ci-dessous représente la topologie du réseau de SMO.

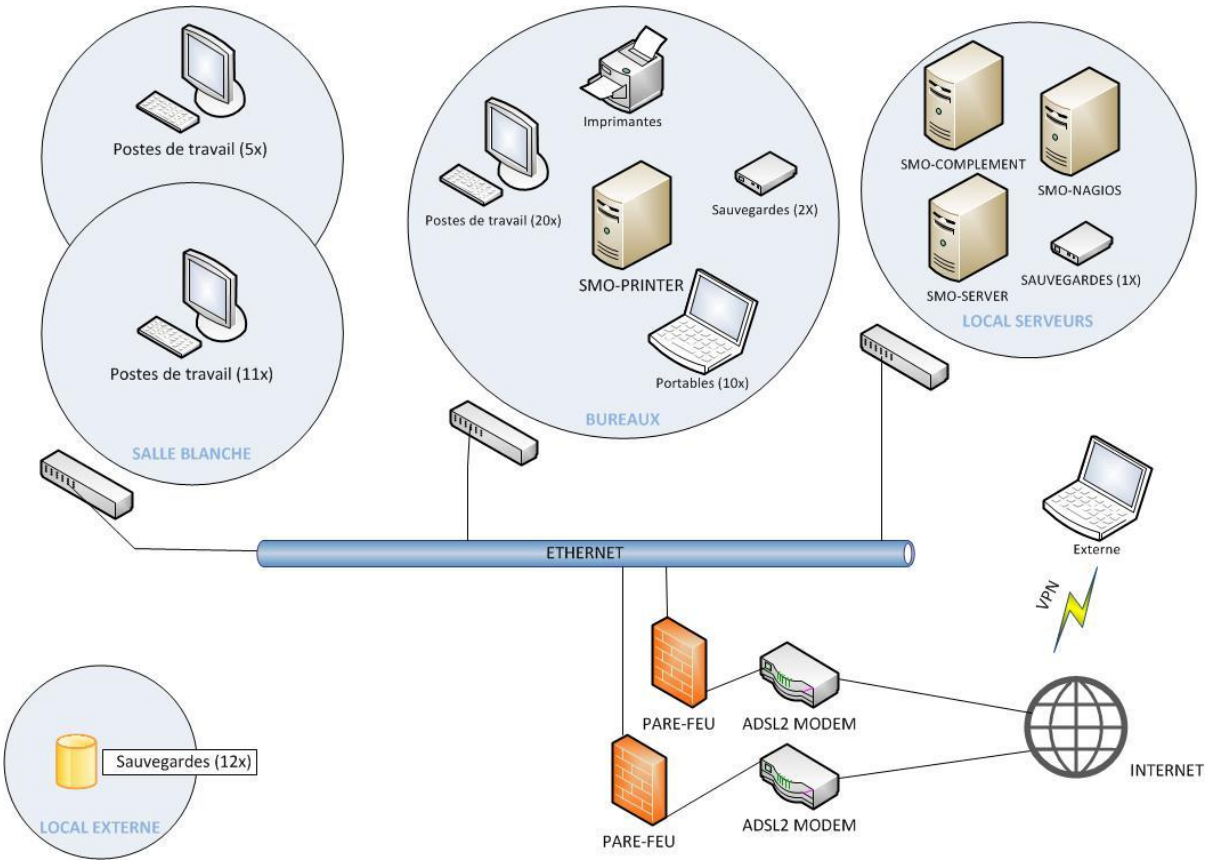


Figure 13 Topologie du réseau SMO

Le SMO-SERVER, est le serveur principal pour les applications DC[36] (Contrôleur de Domaine), DNS[37] (Nom de Serveur de Domaine), DHCP[38] (Protocol réseau), serveur de fichiers, base de données et VPN (réseau privé virtuel).

---

[36] DC Domain Controller est un concept de sécurité réseau windows NT.

[37] DNS Domain Name Server est un protocole pour la gestion des noms de domaine réseau.

[38] DHCP (dynamic host configuration protocol) assure la configuration automatique des paramètres TCP/IP d'une station, notamment en lui assignant automatiquement une adresse IP et un masque de sous-réseau.

Le SMO-COMPLEMENT, est le serveur de fichiers et qui héberge l'application McAfee ePolicy Orchestrator[39].

Le SMO-PRINTER, est le serveur d'imprimantes, de scanners et des mises à jour pour les systèmes d'exploitation Windows.

SMO-COMPLEMENT et SMO-PRINTER ont chacun leur périphérique de sauvegarde.

Les postes de travail et périphériques sont répartis entre les bureaux et la salle blanche avec une dizaine d'ordinateurs portables. Toutes ces machines sont reliées en DHCP sur le réseau LAN Ethernet de SMO et ont un accès à l'internet.

Une fois le diagnostic de SMO fait, nous pouvons maintenant délimiter un périmètre et établir une politique du SMSI.

[39] McAfee ePolicy Orchestrator est un logiciel conçue pour les environnements d'entreprise qui gère de façon centralisée la sécurité des systèmes, des réseaux et des données.

3. Politique et périmètre du SMSI

Ces deux documents ne comportent que quelques paragraphes et peu de contraintes normatives mais leur rédaction doit être mûrement réfléchie car ils sont le point de départ du développement du SMSI. Nous nous sommes par conséquent appuyés sur l'étude préalable et le document *I 8.2.005 Analyse des risques V1.1* pour les rédiger.

3.1 Le périmètre

Le périmètre couvre toutes les activités de SMO définies dans le document *I.4.1.002 Réseau processus V1.0* ci-dessous figure 14.

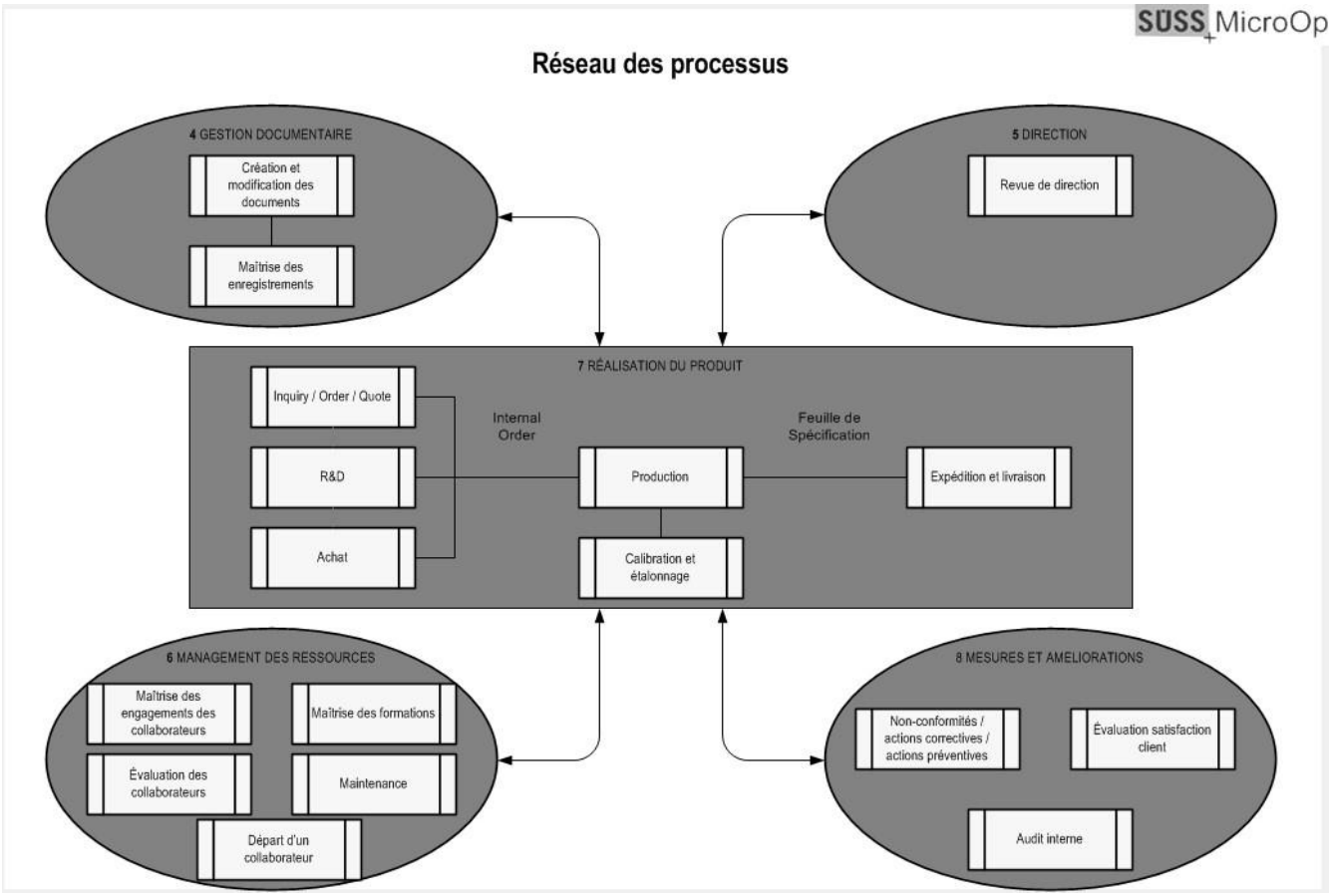


Figure 14 Cartographie des processus de SMO

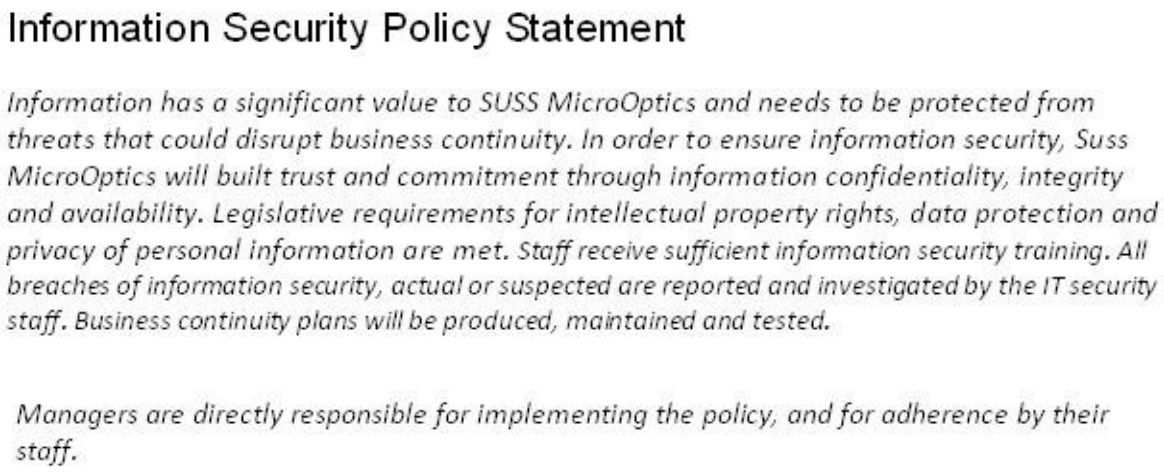
La limite physique du SMSI est matérialisée par l'enceinte des murs des locaux de SMO. La limite du réseau est démarquée par les pare-feu en entrée de la connexion Internet.

3.2 Politique du SMSI

La politique du SMSI de SMO est définie dans le document *I 5.3.001 Security Statement V1.0.* L'extrait de la politique présenté ci-dessous en figure 15 résume les engagements de SMO.

La politique du SMSI rappelle le contexte de risque auquel est soumis SMO et précise la manière par laquelle le SMSI s'intègre dans ce contexte. A titre d'exemple, nous avons vu que SMO participe à la R&D de certains produits de ses clients. En conséquence, SMO s'engage à assurer un niveau de confidentialité, d'intégrité et de disponibilité de l'information satisfaisant et à appliquer les exigences légales, sectorielles et contractuelles en matière de sécurité de l'information.

Cette politique du SMSI est formellement approuvée par la direction qui la valide en la signant. Notons cependant, que cette version du document peut être révisée après l'étape d'appréciation des risques.



Information Security Policy Statement

Information has a significant value to SUSS MicroOptics and needs to be protected from threats that could disrupt business continuity. In order to ensure information security, Suss MicroOptics will built trust and commitment through information confidentiality, integrity and availability. Legislative requirements for intellectual property rights, data protection and privacy of personal information are met. Staff receive sufficient information security training. All breaches of information security, actual or suspected are reported and investigated by the IT security staff. Business continuity plans will be produced, maintained and tested.

Managers are directly responsible for implementing the policy, and for adherence by their staff.

Figure 15 Extrait de la Politique du SMSI de SMO

Une fois la politique et le périmètre établis il convient de sélectionner les bonnes personnes pour conduire le projet et fixer des règles pour la gestion documentaire.

4. Gouvernance et documentation

4.1 Gouvernance

La gouvernance est le terme employé par la norme pour définir la structure de décision. Il existe plusieurs modèles de gouvernance tels que Cobit[40] qui aident à construire une gouvernance cohérente pour les grands comptes. Dans le cas de SMO qui est une PME, nous n'avons pas de structure de décision mais trois responsables choisis sur deux critères. La formation en TI et l'expérience en matière de système de management. La première personne choisie est M. Jocelyn Berger, responsable des technologies de l'information (TI). M. Jocelyn Berger participe aussi à la maintenance du système de management de la qualité ISO 9001 et est par conséquent familier avec tous les processus de l'entreprise et les systèmes de management de l'ISO. En outre, en tant que responsable des TI, il a une forte culture technique ce qui peut favoriser l'adhésion des autres techniciens de l'entreprise au système. La deuxième personne est M. Jean Marc Valentin, pour superviser l'avancement du projet. M. Jean Marc Valentin est « lead auditor » pour les SMSI ISO/CEI 27001. Enfin, je suis spécialiste de la conduite de projet pour la mise en place de systèmes de management et suis responsable de ce projet.

4.2 Documentation

Pour faciliter la maintenance et la lecture du système de management, la norme impose la création d'une procédure décrivant la gestion de la documentation. Il n'existe pas de procédure type mais les points à aborder sont précis. Ils concernent l'approbation, la mise à jour, les versions, la disponibilité des documents. Pour SMO nous avons créé une procédure des exigences relatives à la documentation qui comprend cinq « instructions » intitulées : *Règles typographiques et l'identification des documents*, *Classement informatique*, *File and folder names*, *Most important rules*, et deux procédures, *Création et modification de documents* et *Maîtrise des enregistrements*.

---

[40] Cobit (Objectifs de contrôle de l'Information et des Technologies Associées) est un outil qui permet d'instaurer un langage commun pour parler de la Gouvernance des systèmes d'information tout en tentant d'intégrer d'autres référentiels tels que ISO 9000, ITIL ou ISO/CEI 27001.

L'étape suivante est le délicat processus d'appréciation des risques qui doit aider la direction à sélectionner les actifs qui nécessitent des mesures de sécurité.

5. Appréciation des risques

Nous avons vu précédemment dans les points 1.2.1 Processus d'appréciation des risques et 1.2.2 Méthodes d'appréciation des risques de la partie II, que la séquence des tâches à entreprendre démarrait généralement par une identification des actifs. Dans la phase d'appréciation des risques de SMO nous avons procédé de manière différente pour tenir compte des mesures de sécurité existantes. Nous aurions pu suivre le modèle proposé par la norme ISO/CEI 27005 ou celui de la méthode EBIOS mais nous prenions le risque d'aboutir à une liste de mesures trop éloignées de celles déjà en place. Cette situation peut conduire à l'abandon du projet si dès lors, la direction considère le coût et les contraintes trop élevées. Notre tâche n'est pas de restructurer la sécurité de l'information en place mais de construire un SMSI. Aussi, pour éviter un tel scénario, nous commençons par analyser les écarts entre les exigences de la norme et les mesures de sécurité existantes de SMO. Cette étape appelée « gap analysis » ou SoM (Statement of Maturity) consiste à identifier les actifs sécurisés, les menaces contrées et les vulnérabilités de chacun des actifs.

5.1 SoM (Statement of Maturity)

Le résultat de cette première étape est formalisée dans un tableau intitulé SoM « Statement of Maturity » contenant la liste des 133 mesures de sécurité de la norme ISO/CEI 27001. Pour chacune des mesures de sécurité nous répondons à trois questions.

La première, concerne le niveau de mise en oeuvre « implemented » de la mesure de sécurité. Nous utilisons pour cette mesure le modèle CMMI[41] résumé ci-dessous.

- 0% la mesure de sécurité n'est ni décrite ni appliquée
- 25 % la mesure de sécurité est décrite partiellement et non appliquée et vice-versa

[41] CMMI (Capability Maturity Model Integration) définit une échelle de mesure de la maturité à cinq niveaux, ainsi que les indicateurs nécessaires pour évaluer les activités.

- 50% la mesure de sécurité est décrite partiellement et appliquée partiellement
- 75 % la mesure de sécurité est décrite et appliquée
- 100 % la mesure de sécurité est décrite, appliquée et optimisée

La deuxième, implique une description de la mesure en place, « level situation ».

La troisième, concerne l'évolution de la mesure de sécurité, (planned improvements).

Ci-dessous deux exemples de mesures de sécurité relevées dans le SoM de SMO.

Dans l'exemple tableau 2, nous avons la mesure de sécurité concernant les actifs répertoriés sur le périmètre du SMSI. Chaque actif doit avoir un propriétaire. Chez SMO, cette mesure de sécurité est « décrite et appliquée ». Nous avons pu le vérifier à partir du document *D 6.3.006 Network map*. « L'optimisation » et « l'évolution » de la mesure de sécurité, sera prise en considération lors de la revue de direction si des non-conformités sont constatées.

Tableau 2 Extrait du SoM, propriétaires des actifs

Control Ref. (ISO27001)	Title	Control Description	Implemented	SMO Level Situation	Planned improvements
A.7.1.2	Ownership of assets	All information and assets associated with information processing facilities shall be 'owned' by a designated part of the organization.	75%	Document *D 6.3.008 Network map* , each asset has an owner.	

Le second exemple tableau 3, concerne les sauvegardes. Nous avons constaté que cette mesure de sécurité, bien qu'étant documentée, n'est que « partiellement décrite et appliquée ». La procédure de sauvegarde ne prévoit pas de test des médias utilisés et d'instructions pour la récupération des données. Une mise à jour est prévue pour rendre la procédure conforme au point A.10.5.1 de la norme et compatible avec le DRP (Data Recovery Plan)[42].

[42] DRP (data recovery plan) est un plan de reprise d'activité qui doit permettre à l'organisme en cas de sinistre, de reprendre son activité dans un délai « maximal d'interruption admissible » appelé RTO

Tableau 3 Extrait du SoM, sauvegarde des données

		Control Ref. (ISO27001)	Title	Control Description	Implemented	SMO Level Situation	Planned improvements
To maintain the integrity and availability of information and information processing facilities.		A.10.5.1	Information back-up	Back-up copies of information and software shall be taken and tested regularly in accordance with the agreed backup policy.	50%	*I 6.3.003 Backup Policy* and *I 6.3.006 File server backup* show that requirements regarding backup are fulfilled but there is no instructions for data restoration, backup media are not tested, neither the restoration procedure	Backup and restoration arrangements will be tested to ensure that they meet the requirements of business continuity plan as well as the media used.

Une fois l'analyse des écarts complétée dans le SoM, nous obtenons une vue partielle des risques fondés sur les mesures de sécurité prises par SMO. A ce stade, nous devons vérifier si tous les risques inhérents au SMSI ont bien été identifiés.

Nous allons pour cela construire un tableau « d'appréciation et plan traitement des risques » appelé RATP (Risk Assessment Treatment Plan). Il n'existe pas de modèle type de ce tableau, le responsable de projet est cependant tenu de formaliser sa démarche, l'objectif étant que l'auditeur ou tout autre personne étrangère au SMSI puisse à partir d'un actif sélectionné, évaluer les risques en suivant la méthode proposée.

(recovery time objective) et avec une perte de données « maximale admissible » appelée RPO (recovery point objective).

5.2 RATP (Risk Assessment Treatment Plan)

La figure 16 ci-dessous nous montre le processus d'appréciation des risques sur l'actif « serveur » et la mesure de sécurité qu'on lui applique.

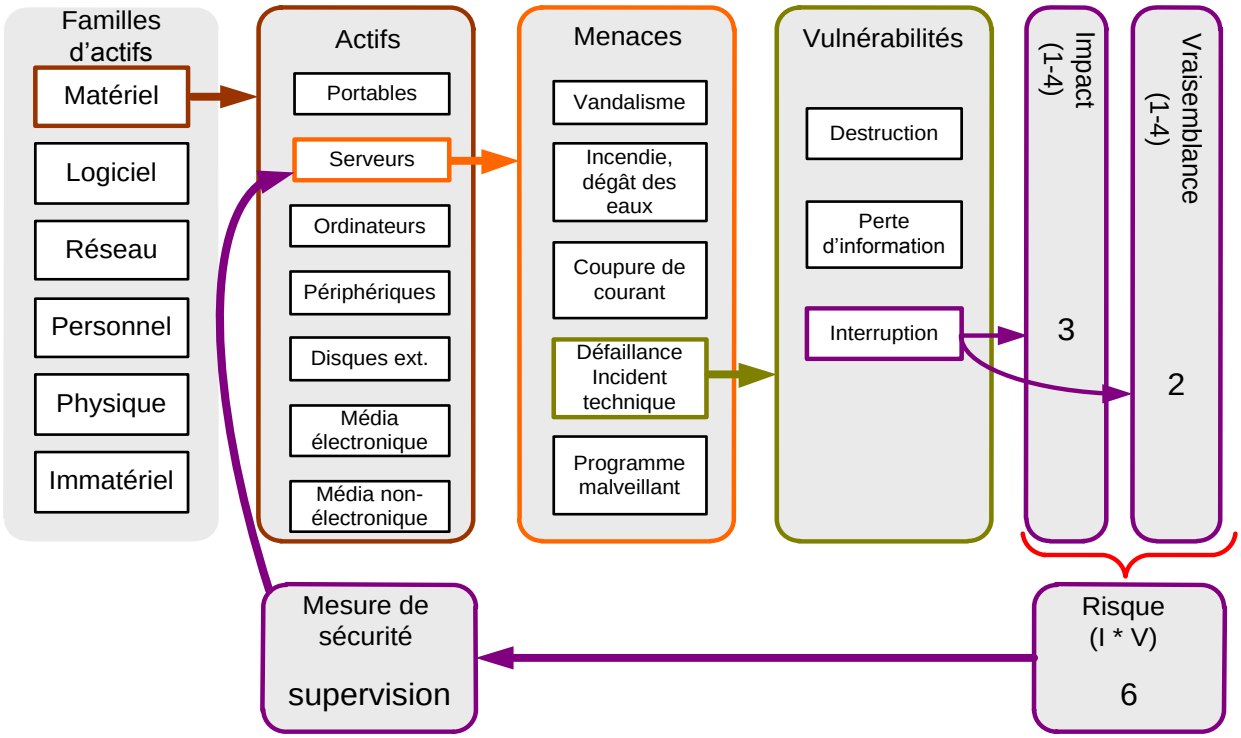


Figure 16 Extrait du RATP

Les deux premières colonnes de notre tableau RATP servent à classer et identifier nos actifs. La première colonne comporte une liste de six familles d'actifs (matériel, logiciel, réseau, personnel, physique, immatériel) dans lesquelles nous avons répertorié dans une seconde colonne, tous les actifs de SMO identifiés sur le périmètre du SMSI. Dans la troisième colonne nous avons défini une liste de menaces potentielles auxquelles peuvent être soumis les actifs. Dans l'exemple figure 16, la menace la plus probable sur les serveurs de SMO correspond à une défaillance ou un incident technique. La « vulnérabilité » serait dans notre cas une interruption du serveur. La « vraisemblance » que cet incident arrive est de « 2 » et « l'impact » qu'aurait une perte de disponibilité du serveur sur les activités de SMO est de « 3 ». Le produit de l'impact par la vraisemblance nous donne « 6 », notre niveau de risque. C'est sur la base du niveau de risque, que la direction décidera d'appliquer une mesure de sécurité. Aussi, la valorisation de l'« impact » et de la « vraisemblance » sont des étapes essentielles de ce processus, car

d'elles dépendront les moyens à déployer pour protéger le système d'information. L'ISO/CEI 27001 n'impose pas d'échelle de mesure spécifique, mais on remarque que la plupart des méthodes d'appréciation des risques ont adopté une échelle de 1 à 4.

L'approche adoptée par SMO pour la mesure de l' « impact », consiste à évaluer les conséquences économiques qu'occasionneraient une perte de confidentialité, d'intégrité et de disponibilité de l'actif concerné, sur son activité. Le tableau 4 ci-dessous montre les critères de valorisation du serveur-SMO.

Tableau 4 Valorisation du Serveur-SMO

Niveau	Confidentialité	Intégrité	Disponibilité
1	Sans conséquences	Sans conséquences	Tolérance > 4 jours
2	Perte < 1 j de CA	Perte < 1 j de CA	4h <tolérance < 4 jours
3	Perte < 1 semaine de CA	Perte < 1 semaine de CA	Tolérance< 4h
4	Perte > un mois de CA	Perte > un mois de CA	Aucune tolérance

CA : Chiffre d'Affaire

Il est à noter que les critères des tableaux de valorisation des actifs manquent de pertinence à la première analyse et doivent être ajustés dans le temps.

La mesure de la « vraisemblance » consiste à déterminer le niveau d'exposition de l'actif face aux menaces, sur une échelle de 1 à 4.

Le niveau de risque s'obtient en prenant la valeur maximum dans notre tableau 4 que l'on multiplie par le niveau de « vraisemblance ». Cette formule peut être affinée en appliquant par exemple un coefficient pour la valeur de confidentialité. Tant que le niveau de risque augmente avec la valeur de l'actif, la pertinence de la formule employée est vérifiée. L'ISO/CEI 27001 encore une fois, ne précise pas comment procéder, mais exige de l'organisme qu'il établisse des critères d'acceptation du risque [15]. La direction a pour cela fixé un niveau au-delà duquel les risques sont réduits, transférés ou évités. Le niveau de risque « acceptable »

tient compte de deux critères, l'impact et le coût. Dans certains cas, un coût trop important conduira à une annulation de la mesure de sécurité.

Afin d'illustrer notre méthode d'appréciation des risques, prenons comme exemple le CPU du serveur-SMO, un pentium4. Nous savons que ces processeurs supportent une température[43] maximale d'environ 70°C. Au-delà de cette limite le serveur peut s'interrompre. Les causes de surchauffe sont variées (arrêt du ventilateur, température du local serveur trop élevée, ajout de nouveaux composants entraînant une surcharge du CPU). En revue de direction, l'analyse de l'appréciation des risques révèlera que le niveau de risque dépasse le seuil « d'acceptation » fixé et conduira le management de SMO à adopter une solution de supervision. Après une étude d'impact et de coût, un premier outil de supervision, MRTG[44] figure 17, permettant le contrôle de la charge du CPU sera choisi.

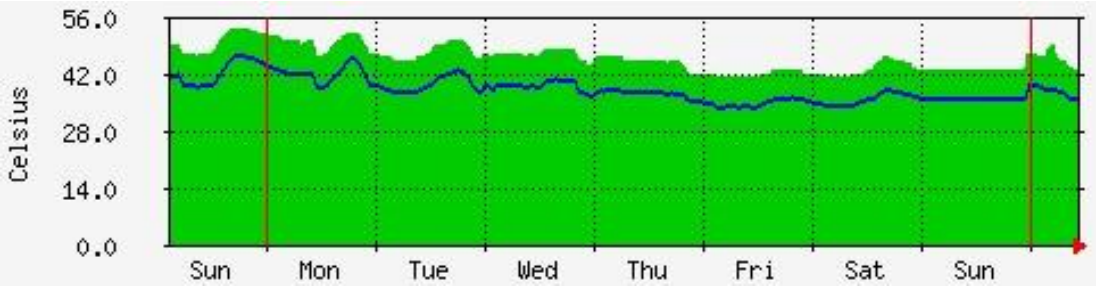


Figure 17 Supervision de la température du CPU (en bleu) avec MRTG

MRTG sera intégré à une plate-forme de supervision, Nagios[45] figure 18. Cette dernière notifiera par courriel à l'administrateur réseau un « Warning » dès que la température du CPU aura dépassé un premier seuil de 60°C. A 70°C, la notification sera d'ordre « Critical ». La supervision du serveur va nous permettre d'amener le risque à un niveau acceptable sans pour autant pouvoir l'écarter

[43] Selon INTEL les processeurs Pentium 4 supportent une température maximum d'environ 70°C.

[44] MRTG (Multi Router Traffic Grapher) est un outil de supervision du réseau qui génère des graphiques. Nous présentons en détails MRTG au point 6.2 du chapitre 6 de la partie IV.

[45] Nagios est une plate-forme logicielle de supervision, nous la présentons en détail dans la partie IV au point 6.1 du chapitre 6.

entièrement, une panne sera toujours possible. Ce risque est appelé résiduel, il doit suivre le même processus d'appréciation si nécessaire[46].

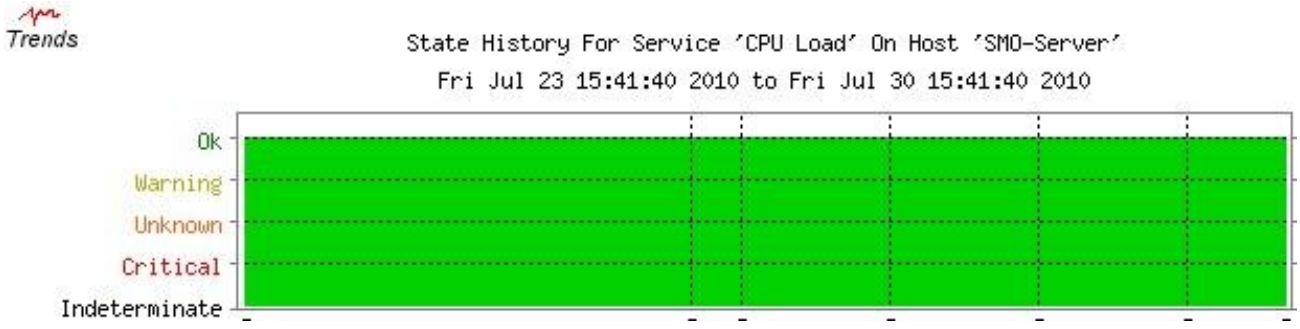


Figure 18 Rapport des alertes concernant la charge CPU sur le serveur-SMO avec Nagios

Au terme de l'appréciation des risques, nous sommes en mesure d'établir le plan de continuité qui suit.

5.3 DRP (Disaster Recovery Plan)

Le DRP ou plan de continuité peut concerner l'ensemble de l'activité de l'organisme. Prenons l'exemple d'un incendie, bâtiments et personnel doivent retrouver leur fonction dans un délai fixé par la direction qui tient compte de l'analyse de risque et de l'impact. Si le délai est court, le plan peut prévoir le repli des collaborateurs sur un nouveau site prévu à cet effet [16].

Pour SMO un tel scénario s'avère compliqué et non rentable. L'activité de l'entreprise reposant essentiellement sur une production réalisée en salle blanche, il serait trop coûteux de maintenir ce type d'installations pour assurer une reprise rapide de la production. Néanmoins, suite à la demande d'un client de SMO, nous avons élaboré un « IT Disaster Recovery Plan ».

Le DRP a été organisé autour de l'analyse de risque et d'impact sur les actifs sensibles tels que les applications et les systèmes des TI.

Le document référencé dans notre système de management *I 6.3.001 IT Disaster Recovery Plan V1.0*, se compose de quatre parties.

---

[46] On constate que le processus d'appréciation des risque est itératif ce qui implique une analyse des risques résiduels.

La première partie : fixe l'objectif du plan.

« The object of this DRP is to restore Critical and Significant applications/systems in a disaster that disables the IT Room or functional area and/or essential equipment supporting the systems or functions. »

La deuxième partie : définit la portée du plan.

« This DRP applies to SUSS MicroOptics IT staff, for protection and restoration of data and services.

Third parties working for IT in any SUSS premise are considered as IT Staff.

This DRP covers business recovery operations. »

La troisième partie : décrit succinctement l'analyse d'impact qui permet d'identifier et de classer les applications et les systèmes des TI nécessitant une attention particulière.

« An initial Business Impact Assessment enables the identification and categorization of applications and systems based on the economic impact of incidents and disasters that result in a denial of access to systems services, and assess the acceptable downtime, that is the length of time business can survive without access to applications, systems, services and facilities. IT Systems or applications are then categorized accordingly as:

- *Critical (less than 4 hours)*
- *Significant (between 4 hours and 4 days)*
- *Low (more than 4 days) ».*

« The Critical & Significant business applications and systems to be recovered are prioritized as described as follows:

DRP Status: 3 options:
- *I = Implemented*
- *NR = Not Required (applicable to systems / applications ranked Low)*
- *TBI = To Be Implemented (IT systems / applications to be implemented should be part of an implementation planning annexed to this DRP)"*

En Annexe 2 et Annexe 3 figurent les tableaux *Applications* et *IT Systems* dans lesquels sont classés les actifs selon le risque et l'impact avec une colonne « *Mitigation* » correspondant aux solutions à apporter en cas de désastre.

La quatrième partie : Annexe 4, est le processus présenté sous la forme de deux datagrammes imbriqués. Le premier, se rapporte à la phase test du DRP qui indique les phases de révision et d'amélioration du processus. Le second, oriente le responsable TI et les différents propriétaires d'actifs concernés en cas de désastres, sur la marche à suivre pour assurer la reprise de l'activité.

Le point suivant, appelée « déclaration d'applicabilité » SoA, est un résumé des besoins du système d'information, il identifie les contrôles choisis et leur niveau d'implémentation.

5.4 SoA (Statement of Applicability)

Le SoA se présente sous la forme d'un tableau dans lequel nous allons simplement cocher les mesures de sécurité que nous décidons de déployer et nous devons apporter une justification pour toutes celles que nous écartons.

En sélectionnant une mesure de sécurité dans le SoA, nous nous engageons à être conforme au modèle PDCA, c'est-à-dire produire une documentation suivant les règles que nous avons définies, générer des enregistrements pour prouver le bon fonctionnement de la mesure de sécurité, et effectuer un contrôle régulier pour vérifier son bon usage.

La liste de mesures de sécurité est essentiellement établie à partir du RATP. Elle complétera celle des clauses[47] 4 à 8 de la norme et des mesures de sécurité qui leur sont directement liées.

[47] Le tableau de l'annexe 00 identifie les correspondances entre les clauses et les mesures de sécurité.

Tableau 5 Mesure "capacity management" du SoA de SMO

		Control Ref. (ISO27001)	Title	Control Description	Implemented	If not chosen, justify.
	A.10.3 System planning and acceptance					
To minimize the risk of systems failures.		A.10.3.1	Capacity management	The use of resources shall be monitored, tuned, and projections made of future capacity requirements to ensure the required system performance.	Nagios - MRTG network ressources monitoring	

Ci-dessus, tableau 5, la mesure de sécurité « capacity planning » que nous avons appliqué à travers la mise en œuvre d'une plate-forme de supervision. Cette étape correspond à la phase « DO » du PDCA, que nous allons maintenant traiter dans la partie qui suit.

PARTIE IV. LA SUPERVISION

Parmi les mesures de sécurité retenues dans le SoA nous avons planifié la supervision des « services » et « hosts »du SMSI avec la mise en place d'une plate-forme logicielle de gestion de la sécurité. Afin de mieux comprendre les outils de supervision déployés, nous présentons dans la partie qui suit, le concept de supervision.

1. Le concept

Une première information nous est donnée par la norme ISO/CEI 27001, au point 10.3.1 capacity management, dimensionnement du système, dont l'extrait est présenté dans la figure 19 ci-dessous.

A.10.3 System planning and acceptance *Objective:* To minimize the risk of systems failures.		
A.10.3.1	Capacity management	*Control* The use of resources shall be monitored, tuned, and projections made of future capacity requirements to ensure the required system performance.

Figure 19 Dimensionnement du système

L'ISO/CEI 27001 impose une supervision et une maintenance des ressources du système d'information pour assurer la « disponibilité » du système. Cependant, la norme n'explique pas comment, ni par quels moyens parvenir à ce niveau de sécurité. L'ISO/CEI 27002 va plus loin en précisant à travers quelques exemples ce qui peut être fait pour obtenir un bon « dimensionnement du système ».

La norme souligne par exemple que l'utilisation des principales ressources, CPU, disque dur, fichiers de sauvegardes, imprimantes et systèmes de communication doit être maîtrisée c'est-à-dire qu'il faut être en mesure d'éviter tout goulot d'étranglement des ressources pouvant présenter un risque pour la sécurité et l'utilisation du système. La supervision de l'utilisation des ressources doit informer de l'état du système pour permettre à l'administrateur d'agir de manière corrective le cas échéant. Quant à l'analyse des informations collectées, elle doit permettre d'intervenir de façon préventive, c'est-à-dire d'anticiper les évolutions nécessaires du système.

Ainsi, la supervision est définie comme étant l'utilisation des ressources pour collecter des informations sur l'état et l'utilisation du système. Le rôle de la

supervision est de servir d'outil pour corriger et optimiser le système d'information et en assurer sa pérennité. De fait, la plupart des entreprises possédant un parc informatique utilisent la supervision pour être plus réactives et proactives [17].

L'OSI[48] avec l'ISO/CEI 7498-4, apporte une définition de l'administration des réseaux, qui permet de mieux situer la supervision en informatique.

2. L'ISO/CEI 7498-4, un cadre pour la supervision des réseaux

La première est appelée « gestion des performances ». Elle porte sur l'aspect communication entre les machines. Le rôle de la supervision est dans ce cas de surveiller la disponibilité des services à travers le contrôle des flux, la latence, le débit. Une fois ces variables collectés, elles sont analysées et comparées aux paramètres fixés préalablement. En cas de dépassement des paramètres fixés, une alerte est envoyée à la ou les personnes désignées pour suivre l'activité du système d'information. Ces alertes peuvent être envoyées sous différentes formes (sms[49], courriels).

La deuxième fonction concerne la « gestion des configurations ». Elle trace et stocke sur une base de données les versions et révisions[50] des informations utilisées par le système telles que le matériel, les logiciels, les données.

La troisième fonction est la « gestion de la comptabilité » qui assure une bonne répartition de l'utilisation des ressources entre les utilisateurs. On peut par exemple fixer des quotas d'utilisation pour chacune des ressources du réseau.

La quatrième fonction, la plus utilisée, concerne la « gestion des anomalies ». Sa tâche est de détecter les anomalies matérielles ou logicielles du réseau, de résoudre si possible automatiquement le problème ou d'alerter la ou les

[48] OSI est un modèle en 7 couches de communications entre ordinateurs proposé par l'ISO.

[49] SMS (short message service) connu en téléphonie mobile, permet de transmettre des messages d'une taille maximale de 160 caractères.

[50] Une révision représente l'instance d'un fichier à un moment donné. Les outils de gestion de version associent généralement à une révision une numérotation de type 1.0, 1.1, 1.2, etc.

personnes concernées. Pour résoudre automatiquement les problèmes, chaque incident est enregistré sur une base de connaissance[51].

La cinquième fonction, « gestion de la sécurité » supervise l'accès aux ressources à partir des droits fixés par la politique de sécurité[52].

La sixième fonction, concerne la « structure de gestion des réseaux ». Elle consiste à installer un agent de supervision sur chaque machine devant être supervisée qui collecte périodiquement et enregistre les informations en local. Si un problème survient, le serveur de supervision est informé pour agir. Ce dernier corrigera le problème en fonction de la nature de l'anomalie. Le serveur de supervision ne reste pas passif, il interroge régulièrement ses agents pour contrôler l'état des machines ou du réseau. Les « objets » stockés dans les bases de données des agents ont été normalisés : ASN.1[53] par l'ISO/CEI tout comme les bases d'information appelées MIB[54] [17].

Nous avons vu les principales fonctions de la supervision. Dans la partie qui suit nous présentons les protocoles CMIP et SNMP, deux solutions sur lesquelles s'appuient les plates-formes logicielles de supervision pour communiquer.

[51] Une base de connaissance regroupe des connaissances spécifiques à un domaine spécialisé donné, sous une forme exploitable par un ordinateur.

[52] La politique de sécurité, ne doit pas être confondue avec la politique du SMSI. Elle contient des directives telles que les règles à suivre pour la création des comptes utilisateurs ou les contraintes sur les flux réseau.

[53] ASN.1 (Abstract Syntax Notation One) est un standard de l'ISO/CEI spécifiant une notation destinée à décrire des structures de données.

[54] MIB (Management Information Base), est un ensemble d'informations structuré sur une entité réseau, par exemple un routeur, un commutateur ou un serveur qui contient une « carte lisible » des OID (Object identifier).

3. Gestion de réseaux

3.1 CMIP

CMIP (Common Management Information Protocol) est un protocole ISO de couche 7 du modèle OSI qui est utilisé avec le protocole CMIS[55] pour la supervision de réseaux hétérogènes. La figure ci-dessous montre les différentes couches du modèle OSI.

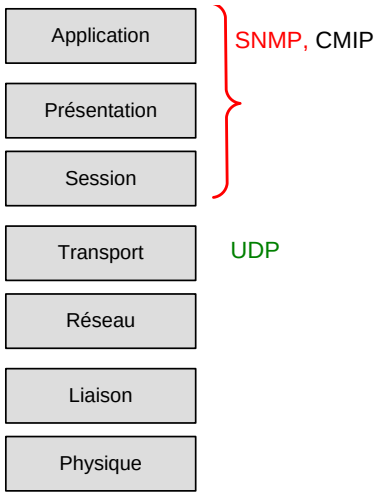


Figure 20 Couches du modèle OSI de l'ISO

CMIP a été proposé par l'ISO en remplacement du protocole SNMP (Simple Network Management Network) moins sophistiqué et moins avancé sur le plan de la sécurité. Mais les spécifications du protocole CMIP ont été jugées trop complexes et trop lourdes à déployer et son implémentation ne s'est faite que dans des secteurs particuliers nécessitant un niveau de sécurité élevé comme par exemple la surveillance radar de l'aviation civile de l'Union Européenne. Actuellement, l'omniprésence de TCP/IP[56] et les évolutions apportées au protocole SNMP, ne laisse plus beaucoup de place au protocole CMIP [17].

[55] CMIS (Common Management Information Services) est un service qui peut être employé par des éléments du réseau pour sa gestion. Il définit le service interface employé par le CMIP.

[56] TCP (Transmission Control Protocol) et IP (Internet Protocol), Le document de référence sur ce sujet est la RFC 1122.

3.2 SNMP

Initialement appelé SGMP (Simple Gateway Monitoring Protocol), le SNMP[57] (Simple Network Management Protocol) a été développé par l'IETF[58]. Depuis la sortie de la première version en 1990, SNMP a évolué, quelques améliorations ont été apportées au protocole, telles qu'une couche sécurité avec cryptage pour la dernière version SNMPV3 sortie en 2002. L'idée directrice était d'en faire un protocole simple qui tienne compte de la charge du transport des données de supervision sur les éléments du réseau. SNMP est un protocole de communication compatible avec les plates-formes hétérogènes. Il gère les éléments du réseau, supervise et diagnostique les problèmes. TCP/IP étant très répandu, le protocole SNMP est rapidement devenu incontournable pour les administrateurs en charges d'établir la supervision et l'administration des équipements de leur réseau tels que les commutateurs, les routeurs, les serveurs et les imprimantes [18].

Comme son nom l'indique, le protocole SNMP se veut simple et rapide, il se situe sur la couche 7 du modèle OSI et s'appuie sur le protocole UDP[59] (User Datagram Protocol) pour adresser ses requêtes.

SNMP fonctionne sur un modèle client-serveur avec d'un côté un serveur superviseur NMS (Network Management Station) et de l'autre les serveurs sur lesquels sont installés les agents SNMP. Les agents SNMP informent de l'état des « nœuds » (équipements gérés) qui contiennent des « objets[60] » gérables (informations matérielles, statistiques de performance, paramètres de configuration). Ces derniers sont classés dans la base d'information de gestion MIB [18].

---

[57] RFC1157 (requests for comments), aspects techniques de la première version de SNMP. Notons qu'un standard IETF constitue par défaut un RFC.

[58] IETF (Internet Engineering Task Force) est un groupe informel, international, ouvert à tout individu, qui participe à l'élaboration de standards pour Internet. L'IETF produit la plupart des nouveaux standards d'Internet.

[59] UDP, chaque trame possède une adresse source et une adresse destination qui permet à SNMP et aux autres protocoles de niveau supérieur d'adresser leurs requêtes.

[60] Rappelons que l'OID est l'identifiant de l'objet.

Ci-dessous la figure 21 montre une vue générale de la communication entre le serveur superviseur et les agents SNMP.

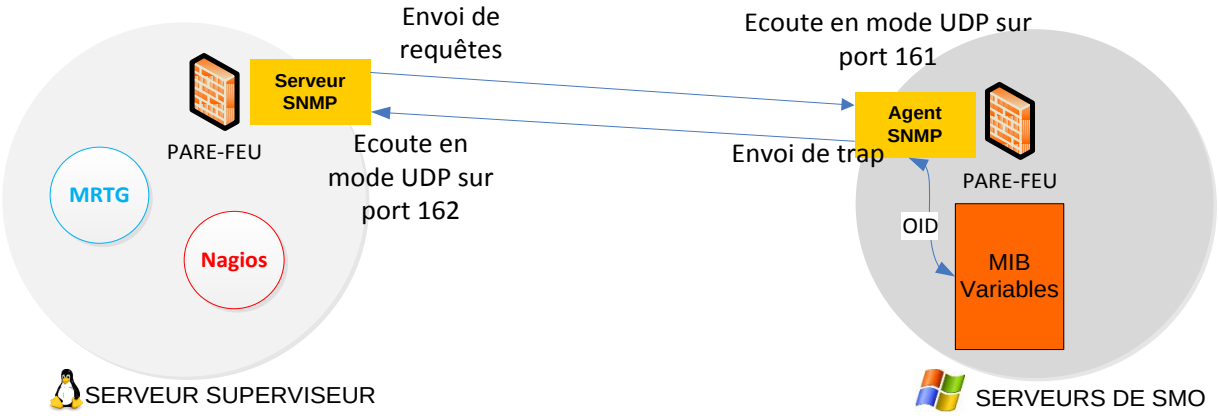


Figure 21 Fonctionnement de SNMP sur les serveurs de SMO

Le protocole SNMP supporte trois types de requêtes, Get, Set, Trap. Pour la gestion du réseau SNMP utilise les opérations suivantes :

Une requête est envoyée par le superviseur sur un des agents via le port 161 UDP, elle peut être de quatre types :

- GetRequest : recherche d'une variable de la MIB sur un agent
- GetNextRequest : permet de recevoir le contenu de l'instance qui suit l'objet nommé dans la MIB.
- GetBulk : recherche un ensemble de variables regroupées
- SetRequest : modifie la valeur d'une variable sur un agent

L'agent traite la requête et envoie une réponse via le même port, GetResponse avec la valeur demandée. En cas d'erreur, la réponse de l'agent est accompagnée d'un des codes suivants :

- NoAccess : accès non autorisé
- WrongLength : erreur de longueur
- WrongValue : erreur de valeur
- WrongType : erreur de type
- WrongEncoding : erreur d'encodage

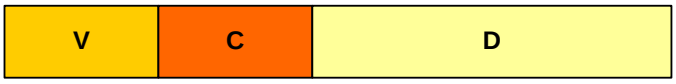
- NoCreation : objet inexistant
- ReadOnly : seule la lecture est autorisée
- NoWritable : interdiction d’´écrire
- AuthorisationError : erreur d’autorisation

Si un incident ou événement survient sur la ressource supervisée une alarme est envoyée par l’agent. Le message envoyé au serveur superviseur est de type « Trap » ou « Inform » et passe par le port 162 UDP.

Les alarmes peuvent prendre les formes suivantes :

- ColdStart(0) : redémarrage à froid du système
- WarmStart(1) : redémarrage à chaud du système
- LinkDown(2) : le lien réseau n'est plus opérationnel
- LinkUp(3) : le lien réseau est de nouveau opérationnel
- AuthentificationFailure(4) : tentative d'accès à l'agent avec un mauvais nom de communauté
- EGPNeighboorLoss(5) : la passerelle adjacente ne répond plus
- EntrepriseSpecific(6) : alarme spécifique aux constructeurs

Le datagramme UDP de SNMP se construit comme suit,



- V, version 1, 2 ou 3 de SNMP
- C, type de communauté[61], publique ou privée
- D, données, requêtes : (Get, GetRequest, trap).

La figure ci-dessous nous montre la communication entre le serveur superviseur SNMP et ses agents.

[61] Il est possible de paramétrer des communautés (groupes de sécurité) qui ont différents privilèges, lecture, lecture et écriture ou lecture sur certaines branches uniquement. Le groupe est identifié par un mot de passe appelé « communauté publique ou privée »

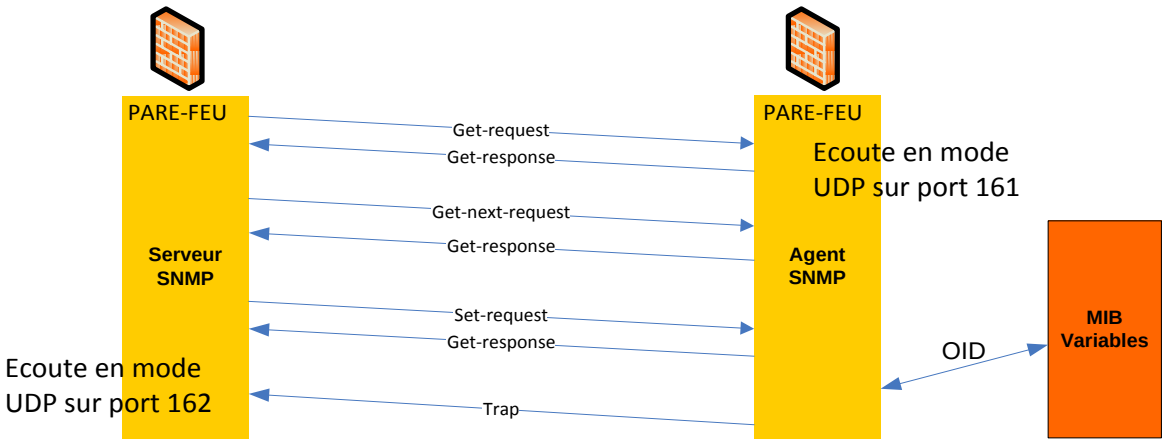


Figure 22 Communication entre le superviseur et les agents SNMP

Notons qu'aujourd'hui, la plupart des logiciels de supervision sous licence GNU/GPL[62] s'appuient sur le protocole SNMP pour interroger les services et « hosts », alerter et produire des graphiques permettant aux administrateurs de suivre l'évolution de leurs équipements.

Nous pouvons dire que l'ISO, le CEI et l'IETF ont apporté un format commun des données, un modèle d'administration pour les adresser et des protocoles pour les échanger. Enfin, cette partie nous a permis de comprendre le fonctionnement du protocole SNMP et de voir quels avantages un administrateur de réseau peut avoir en adoptant cette solution. Dans le point qui suit nous expliquons comment a été installée et configurée la version 2c de SNMP [19].

4. Mise en oeuvre de SNMP

La première tâche consiste à installer les librairies pour SNMP, le superviseur SNMP, le démon et les MIBs. L'installation de fait à partir le gestionnaire de paquets « apt-get [63]».

```
sudo apt-get install libsnmp-base libsnmp-perl libsnmp9-dev snmp
snmpd tkmib
```

[62] GNU/GPL Licence publique générale GNU, ou GNU General Public License est une licence qui fixe les conditions légales de distribution des logiciels libres du projet GNU. Chaque personne qui adhère aux termes et aux conditions de la GPL a la permission de modifier le travail, de l'étudier et de redistribuer le travail ou un travail dérivé.

[63] Apt-get gère les paquets, sa principale fonction est de résoudre les dépendances automatiquement.

L’installation se fait en deux phases, la première concerne le serveur superviseur (Nagios), la deuxième les agents des serveurs Windows NT de SMO. Il est à noter que les configurations de tous les serveurs SMO doivent être identiques.

4.1 Installation de SNMP sur le serveur superviseur

Nous configurons l’authentification en définissant la « communauté » en mode public ou privé puis attribuons les privilèges en écriture et lecture pour les serveurs du LAN SMO.

La configuration de SNMP se trouve dans les fichiers snmpd.conf et snmpd.

On modifie les fichiers snmpd.conf et snmpd.

```
#Extrait du fichier de configuration /etc/snmp/snmpd.conf

#Autorisation des requettes


com2sec local localhost public

com2sec localnetwork 192.168.0.0/24 public

#lier la sécurité avec le groupe

group MyRWGroup_1 v2c local

group MyRWGroup_2 v2c localnetwork

view all included .1 80

#création d'une vue avec les droits du groupe

access MyRWGroup_1 « «  any noauth exact all none none

access MyRWGroup_2 « «  any noauth exact all none none
```

```
#Extrait du fichier de configuration /etc/default/snmpd
SNMPDOPTS='-Lsd -Lf /dev/null -u snmp -I -smux -p /var/run/snmpd.pid'
```

On redémarre le service.

```
sudo /etc/init.d/snmpd restart
```

On vérifie que le service SNMP a démarré.

```
sudo ps -aux ¦ grep snmp
```

Enfin, on vérifie que le serveur soit bien configuré avec la commande snmpwalk[64]. Cette commande permet de lire toutes les valeurs de l'agent SNMP.

```
sudo snmpwalk -v 2c public -c 192.168.169.7
```

4.2 Installation des agents SNMP sur les serveurs SMO

La première tâche sur les serveurs Windows NT SMO est de configurer via le service réseau du panneau de configuration le service SNMP (agent).

On configure l'authentification du service SNMP (public ou privée) sur le serveur Windows NT pour permettre au serveur superviseur d'entrer en communication.

Les commandes suivantes permettent d'établir une communication sur les serveurs SMO.

```
sudo snmpwalk -v 2c public -c 192.168.xxx.x

sudo snmpwalk -v 2c public -c 192.168.xxx.x

sudo snmpwalk -v 2c public -c 192.168.xxx.x
```

A ce stade de l'installation, le protocole SNMP est en service. Il va servir de protocole de communication à notre plate-forme logicielle de supervision.

Dans la partie qui suit nous présentons les architectures et l'état de l'art des logiciels de supervision avant d'exposer les solutions retenues.

5. Les logiciels de supervision

5.1 Types de déploiement

Les logiciels de supervision peuvent être déployés de trois façons différentes.

5.1.1 Déploiement centralisé

Le déploiement centralisé consiste comme son nom l'indique à assurer le processus de supervision à partir d'une machine. Cette configuration comporte

[64] Snmpwalk est une commande Unix qui permet d'interroger une machine pourvue d'un agent SNMP à travers un réseau. La commande obtient des informations successivement en utilisant la commande GETNEXT du protocole SNMP, permettant ainsi de consulter une partie de l'arbre formé par les OID de la MIB.

quelques inconvénients. Le premier est la panne de la machine superviseur entraînant l'arrêt de la surveillance des éléments du réseau. Un autre inconvénient est que le processus de supervision est lourd, et exige une machine robuste pour traiter la supervision. Enfin, concentrer l'ensemble des requêtes sur une machine accroît fortement le trafic en provenance de celle-ci.

5.1.2 Déploiement hiérarchique

Le déploiement hiérarchique est plus tolérant vis-à-vis des pannes que le déploiement centralisé mais il est plus difficile à mettre en œuvre. Les différents serveurs utilisés devant être synchronisés, les temps de réponse sont plus longs. Chaque segment du réseau est supervisé par un serveur qui dialogue avec un serveur central parent. On peut avoir plusieurs niveaux hiérarchiques, chacun étant client et serveur de supervision.

5.1.3 Déploiement distribué

Enfin, le déploiement distribué, est une combinaison des deux précédentes configurations. Les serveurs possèdent une base de données « d'objets ». Ils peuvent s'échanger leurs données ce qui permet par exemple, de spécialiser un serveur sur un type de supervision (alarme, performances).

La mise en œuvre d'une plate-forme logicielle de supervision passe de fait par l'application d'un de ces modèles de déploiement.

La partie qui suit est consacrée aux outils et plates-formes logicielles de supervision réseau.

5.2 Outils de surveillance du réseau

Ces outils ne peuvent être comparés aux plateformes logicielles de supervision. Cependant, il est possible de développer leurs fonctionnalités de base à l'aide de langages scripts pour obtenir des résultats intéressants [18].

A titre d'exemple, nous avons sélectionné deux outils parmi les plus utilisés aujourd'hui par les plates-formes logicielles de supervision.

5.2.1 Ping

Le plus simple des outils est Ping [20]. Il peut sur le protocole CMIP, envoyer une requête « Echo », attendre sa réponse « EchoReply » pour, par exemple :

- Mesurer la latence entre deux serveurs.
- Savoir si un serveur est accessible, sinon recevoir un code d'erreur CMIP sous forme d'alerte à l'administrateur.

5.2.2 Traceroute

Traceroute mesure les problèmes de routage et de congestion sur le réseau internet. Il permet de suivre le chemin parcouru par un paquet à travers les nœuds du réseau internet. Notons que selon la version de Traceroute, différents protocoles (UDP, TCP ou CMIP) sont utilisés en couche transport ou application [21].

5.2.3 Netstat

Netstat (Network Statistics) peut afficher des statistiques sur presque tous les protocoles réseaux tels que, Ethernet, IP, CMIP, TCP, UDP. Il affiche également des informations sur les tables de routage, connexion des ports, paquets et interfaces réseaux [22].

5.2.4 RRDTool

RRDTool, est un outil de gestion de base de données s'appuyant sur le système RRD[65] (Round-Robin Data Base). RRDTool est sous licence GNU/GPL. Il est considéré comme une évolution de MRTG[66] et en reprend les nombreuses options graphiques en les améliorant. RRDTool est surtout utilisé pour la sauvegarde de données cycliques, chronologiques et le tracé de graphiques. Il permet aussi de superviser les données du serveur, comme la bande passante ou la température d'un CPU. RDDTool est aujourd'hui un outil utilisé par la plupart des plates-formes logicielles de supervision [23].

Il existe encore des dizaines d'autres outils comme les programmes destinés à utiliser le protocole SNMP tels que snmpget, snmpinform, snmpset. Ces requêtes sont souvent employées dans les scripts qui stockent les valeurs pour envoyer des alertes. Un autre outil souvent utilisé est IPERF. Il permet de mesurer les

[65] Le round-robin est un jeu de parcs : un tourniquet. C'est l'idée intuitive derrière l'algorithme, chaque processus est sur le tourniquet et ne fait que passer devant le processeur, à son tour et pendant un temps fini.

[66] MRTG (Multi Routeur Traffic Grapher) est un outil de supervision. Il fait partie des solutions retenues pour la supervision chez SMO. Il est présenté au point 6.2.

différentes variables d'une connexion sur le réseau. Tous ces outils améliorent les performances et élargissent les fonctionnalités des plates-formes logicielles de supervision.

Dans la partie qui suit nous présentons quatre plates-formes logicielles de supervision parmi les plus répandues et reconnues.

5.3 Plates-formes logicielles de supervision

5.3.1 Cacti

Cacti est un logiciel sous licence GNU/GPL qui permet de mesurer la performance des serveurs et du réseau en gérant nativement SNMP. Il produit des graphiques reposant sur une interface Web basée sur PHP et MySQL et utilise le moteur RRDTool pour collecter les statistiques. Dans sa version de base, Cacti ne propose pas la gestion des incidents, ni les alertes en cas de panne matérielle ou dysfonctionnement logiciel. Il offre une vision dans le temps de l'évolution d'indicateurs tels que le volume des données sur les disques durs, la latence ou le trafic réseau. L'intérêt de ce logiciel réside essentiellement dans sa partie graphique. Contrairement aux autres logiciels de supervision tels que Nagios[67], Cacti génère des images dynamiques à partir des fichiers RRDTool. Un autre avantage réside dans le fait qu'il est possible de partager ses modèles (templates) avec d'autres utilisateurs [24]. Enfin, l'installation de Cacti est simple et ne présente pas de difficultés.

5.3.2 Zabbix

ZABBIX est un logiciel de supervision sous licence GNU/GPL disponible sur de nombreuses plateformes. Il permet via SNMP, de surveiller les services et composants du réseau, les protocoles, les ports, les processus, les ressources matérielles, etc. par des tests de connexion simples (ping, test de ports). On peut paramétrer la fréquence des vérifications, personnaliser les alertes et créer des groupes d'utilisateurs ou groupes d'hôtes. Zabbix génère aussi des graphiques avec une interface Web PHP. Son moteur écrit en langage C, collecte les données

[67] Nagios est le logiciel de supervision que nous mettons en place pour SMO, voir le point 6.1.

et génère les alertes. Son interface Web basée sur PHP, permet l'administration des graphiques et la consultation des données. Zabbix est le plus aboutis des récents logiciels de supervision. Il se place en concurrence directe avec Nagios avec quelques avantages sur son rival comme l'intégration du RRDTool pour les graphiques dynamiques. On peut conclure en soulignant que les fonctionnalités avancées de ce logiciel n'ont pas altéré sa simplicité d'utilisation, d'installation et de paramétrage [25]. En outre, le développement de Zabbix est soutenu par une entreprise et une communauté grandissante, lui assurant une évolution durable.

5.3.3 OpenNMS

OpenNMS est distribué sous licence GNU/GPL. C'est un logiciel multiplateforme (Linux, Solaris, Mac OS, MS Windows) de supervision fonctionnant sur un moteur Jetty[68] avec une base de données PostgreSQL et un serveur web. Il offre toutes les possibilités d'un logiciel de supervision : surveillance des services et serveurs, collecte des données via le protocole SNMP, évaluation des performances, gestion des événements, des alarmes et des notifications. OpenNMS génère aussi des graphiques à partir de polling SNMP[69] ainsi qu'une représentation graphique des équipements supervisés. OpenNMS est une des plateformes de supervision les plus complètes [26].

5.3.4 Centreon

Centreon est un logiciel commercial développé sur la base de Nagios. Ses développeurs ont souhaité apporter une interface moins austère que celle proposée par Nagios en y ajoutant quelques fonctionnalités. L'interface se compose de six vues, une page d'accueil « home » reprenant le « tactical overview » de Nagios qui offre une vue globale des incidents ou événements du réseau ainsi qu'un accès aux statistiques des performances du moteur et ses composants, une page « monitoring » avec plusieurs vues disponibles, une page « views » qui offre un accès à tous les graphiques par un menu arborescent ainsi

[68] Jetty est serveur http et un moteur de servlet basé sur le langage JAVA.

[69] Le processus polling est une vérification active des valeurs par SNMP contrairement aux traps qui sont une vérification passive des valeurs par SNMP.

qu'à une cartographie du réseau, une page « reporting » sous forme de tableau de bord, une page « configuration » et « administration » pour la gestion des accès utilisateurs. Le principal avantage de Centreon est qu'il a hérité des meilleures fonctionnalités de Nagios en sachant les faire évoluer [27]. Son développement est assuré par une entreprise ce qui implique un coût pour acquérir sa licence. Enfin, l'ajout de fonctionnalités l'ont rendu plus lourd que Nagios en termes de ressources matérielles.

En résumé, on trouve les mêmes fonctionnalités sur la plupart de ces plates-formes logicielles de supervision sauf pour Cacti qui ne fait pas réellement de supervision. Evaluer ces plateformes est de fait difficile, il s'agit d'étudier non seulement les caractéristiques de chaque logiciel mais aussi toutes les contraintes et avantages d'intégration. Le tableau comparatif présenté en Annexe 5 peut servir de point de départ pour une première réflexion.

Dans le chapitre suivant nous présentons les solutions retenues et leur mise en œuvre

6. Mise en œuvre de la plate-forme Nagios – MRTG

6.1 Présentation de Nagios

Nagios est la plate-forme logicielle de supervision sous licence GNU/GPL la plus connue. Ethan Galstad est à l'origine de la première version, développée sous le nom de NetSaint en 1999. Depuis, Nagios est utilisée par de grandes entreprises et institutions telles que, Air France, le CNRS et l'Education Nationale. Nagios est bien documenté avec un grand nombre de sites et forums Web qui lui sont dédiés. Parmi les solutions étudiées, Zabbix et Centreon répondaient amplement aux besoins du projet de SMO. Centreon est plus complet que Nagios et son installation plus facile mais le logiciel est sous licence commerciale. Zabbix est aussi complet avec une belle interface graphique offrant de nombreuses vues sur l'état des services mais il est moins documenté que Nagios. Pour finir, Nagios nous a paru comme étant la solution la plus intéressante à mettre en oeuvre.

Nagios étant à l'origine de la plupart des logiciels de supervision, son installation et sa configuration nous permettra de mieux comprendre les mécanismes des plates-formes logicielles de supervision. Dans la figure 23 ci-dessous, une vue de l'évolution de Nagios de 1999 à aujourd'hui.

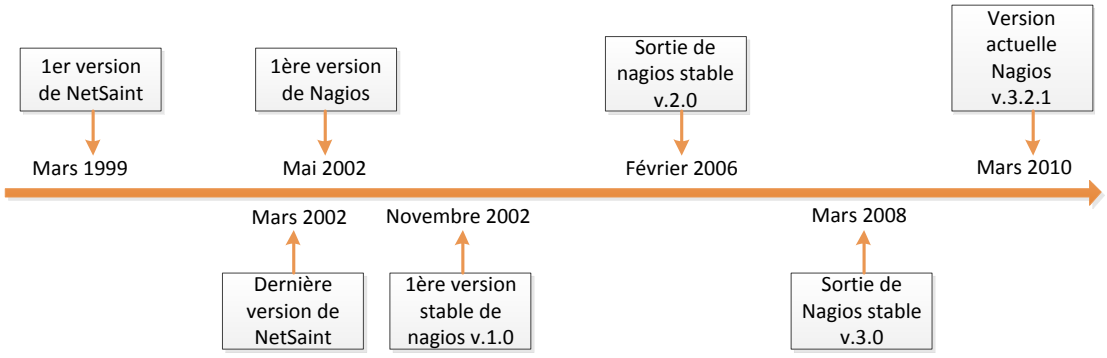


Figure 23 Evolution de Nagios

La version stable actuelle de Nagios est la 3.2.1. Nagios fonctionne sur un serveur Linux ou Unix à partir duquel il peut superviser l'ensemble des services du réseau.

La première fonction de Nagios consiste à gérer l'ordonnancement des vérifications et les actions à prendre sur les incidents (alertes, escalades). Il existe une pléiade de fonctionnalités maintenues par une large communauté de développeurs [28]. On peut ainsi choisir des plugins existants ou en programmer de nouveaux en fonction de ses besoins. Ci-dessous un résumé des fonctionnalités les plus courantes.

- Surveillance des services du réseau (SMTP, POP, http, Ping, SNMP)
- Surveillance des ressources des serveurs (mémoire, disques durs, CPU)
- Surveillance de données comme la température, la luminosité, humidité d'une salle à l'aide de sondes.
- Surveillance des files d'attente d'impression.
- Notification d'alerte paramétrable.
- Répartition de la supervision entre les administrateurs.
- Définition des gestionnaires d'évènements qui s'exécutent pour une résolution proactive des problèmes.
- Journal avec rotation automatique des fichiers pour un historique des évènements.

- Surveillance à partir d'un navigateur Web.
- Par un accès sécurisé l'administrateur peut via un VPN[70] superviser les ressources.
- Procédures d'alertes et d'escalades[71].
- Cartographie du système d'information supervisé.
- Edition de rapports pour analyse.
- Ces fonctionnalités permettent à l'administrateur d'intervenir immédiatement sur un « service » ou un « host » montrant un problème.

En résumé, Nagios permet de satisfaire la plupart des besoins de supervision, hormis la métrologie[72]. Il n'est pas possible de faire de la supervision sans parler de métrologie. Aussi, en complément de Nagios nous avons intégré un des outils les plus connu en la matière, MRTG (Multi Router Traffic Grapher).

6.2 Présentation de MRTG (Multi Router Traffic Grapher)

En 1994, Tobias Oetiker, cherche à contrôler sur un routeur la charge du trafic d'une ligne de 64Kbit/s et développe pour cela un petit programme[73]. Très vite, les possibilités de supervision de cet outil ont été étendues à tous les éléments actifs du réseau grâce à SNMP ainsi qu'à tous les composants matériels des serveurs (mémoire, volume du disque dure, CPU).

Comme son nom l'indique, « Multi Router Traffic Grapher » génère à partir des données collectées, des graphiques au format PNG[74] sur un navigateur Web.

---

[70] VPN (Virtual Private Network) est une extension des réseaux locaux et préserve la sécurité logique que l'on peut avoir à l'intérieur d'un réseau local.

[71] La gestion des escalades concerne le routage d'une alerte sur un groupe ou personne concernée par l'alerte. Cela peut consister à faire remonter l'alerte dans la hiérarchie de l'entreprise. Autre exemple, une alerte peut être envoyée à un autre « contact » si elle n'est pas acquittée.

[72] La métrologie est le graphage des données collectées.

[73] En annexe 9, un extrait du programme écrit par Tobias Oetiker en langage Perl.

[74] PNG (Le Portable Network Graphics) est un format ouvert d'images numériques, qui a été créé pour remplacer le format GIF. Il est adapté pour publier des images simples comprenant des aplats de couleurs.

MRTG cherche les données à partir d'un script Perl via le protocole SNMP et les envoie à un programme écrit en langage C[75], qui les stocke et en génère des graphiques. L'état des éléments du réseau est affiché avec une fréquence paramétrée de cinq minutes [29].

Le choix du MRTG a été motivé pour diverses raisons. C'est un logiciel sous licence GNU/GPL comprenant une bonne documentation et une large communauté de développeurs qui assure l'évolution du logiciel. En outre, l'exploitation du MRTG nécessite peu de ressources matérielles et il s'intègre parfaitement à Nagios [30]. Les graphiques qu'il peut générer serviront d'outil de mesure pour les indicateurs ce qui donnera à la direction de SMO un support d'analyse pour fixer de nouvelles priorités et octroyer de nouvelles ressources pour assurer la pérennité du système d'information. A titre d'exemple, les indicateurs peuvent concerner les attaques, les sauvegardes, la perte ou surcharge du réseau. La figure 24 ci-dessous, montre une attaque par DoS[76] sur le système.

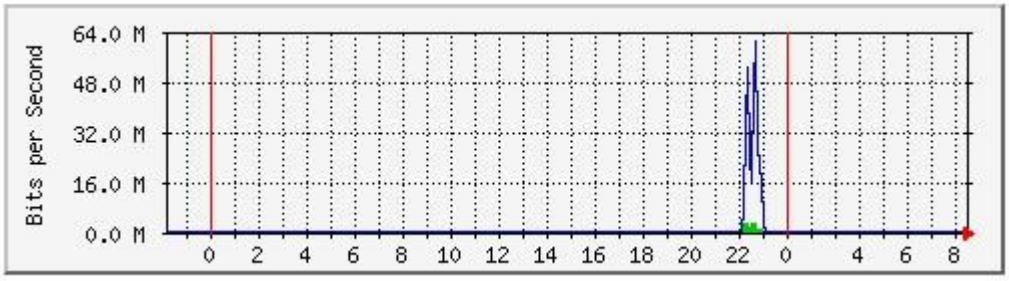


Figure 24 DoS attaque sur le réseau

Nous avons présenté les différentes fonctionnalités et complémentarités de Nagios et de MRTG. Dans le point suivant, nous décrivons leur mise en oeuvre.

[75] En annexe, un extrait du programme écrit par Dave Rand en langage C.

[76] DoS (Denial of Service) est une attaque ayant pour but de rendre indisponible un service, d'empêcher ses utilisateurs de l'utiliser. Il peut s'agir de l'inondation d'un réseau, la perturbation des connexions entre deux machines ou l'obstruction d'accès à un service.

6.3 Pré-requis d'installation des composants

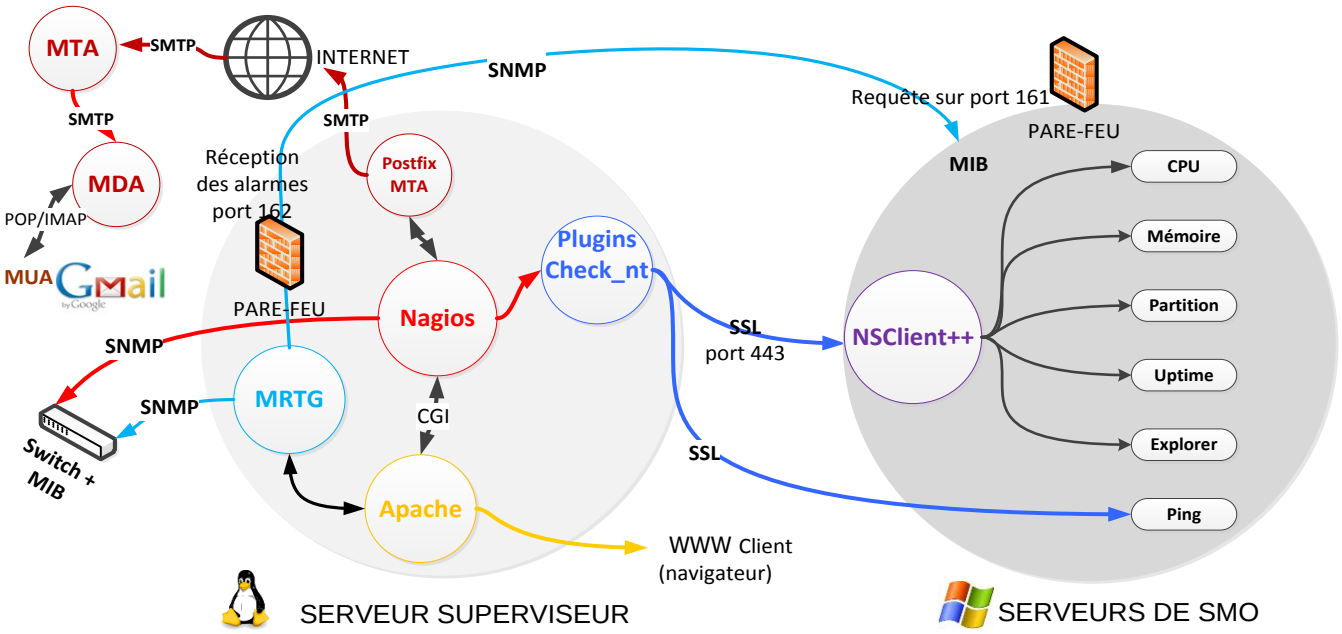


Figure 25 Vue générale de la mise en oeuvre de Nagios et MRTG

Comme le montre la figure 25 ci-dessus, la mise en place des logiciels de supervision Nagios et MRTG, nécessite l'installation de pré-requis tels que Apache[77], Perl[78], Php[79], GCC[80] et les librairies Apache2 et GD-2[81].

Pour l'installation de ces paquetages on utilise leur configuration par défaut. Apache, est le serveur Web basé sur les CGI[82] qui sert d'IHM (Interface Homme Machine).

---

[77] Apache est un logiciel de serveur HTTP (serveur web).

[78] Perl est un langage de programmation reprenant des fonctionnalités du langage C et des langages de scripts.

[79] Php (Hypertext Preprocessor) est un langage de scripte utilisé pour la création de page Web dynamique.

[80] GCC (GNU Compiler Collection) est le compilateur capable de compiler divers langages de programmation, (C, C++, Objective-C, Java, Ada et Fortran).

[81] GD (Gift Draw) est une libraire qui sert à manipuler dynamiquement des images au format (gif, png, jpeg, wbmp, xbm, xpm).

```
sudo apt-get install apache2 apache2.2-common build-essential

libgd2-xpm-dev GCC-4.0 perl - dev php5-commun php5 libapache2-mod-
php5
```

Installation de la configuration apache pour l'interface Web de Nagios.

```
sudo make install-webconf

sudo /usr/bin/install -c -m 644 sample-config/httpd.conf
/etc/apache2/conf.d/nagios.conf
```

Dans le fichier de configuration d'Apache2 il est possible de configurer tous les paramètres (ports, serveur, etc.). Pour SMO seule la configuration de l'authentification [4] a été paramétrée, pour le reste nous avons conservé la configuration définie par défaut.

Création du compte nagiosadmin pour l'authentification de l'interface Web. L'accès par authentification s'effectue en entrant un nom d'utilisateur et un mot de passe. Cet accès doit être restreint car il permet de redémarrer un service ou d'acquitter une alarme.

```
Sudo htpasswd -c /usr/local/nagios/etc/htpasswd.users nagiosadmin
```

Redémarrage du démon Apache

```
Sudo /etc/init.d/apache2 restart
```

Une fois les pré-requis installés, les composants de MRTG et de Nagios peuvent être mis en place.

6.4 Mise en œuvre de MRTG

Le fonctionnement de MRTG consiste à interroger les bases d'informations MIB des équipements [29]. Une fois recueillies, les informations sont stockées dans le fichier mrtg.cfg puis utilisées pour générer des graphiques qui sont placés dans un répertoire de page Web (index.html).

---

[82] CGI (Common Gateway Interface) ou (Interface passerelle commune), est une interface normalisée utilisée par les serveurs http. CGI est le standard qui indique comment transmettre la requête du serveur Web au programme et comment récupérer la réponse générée.

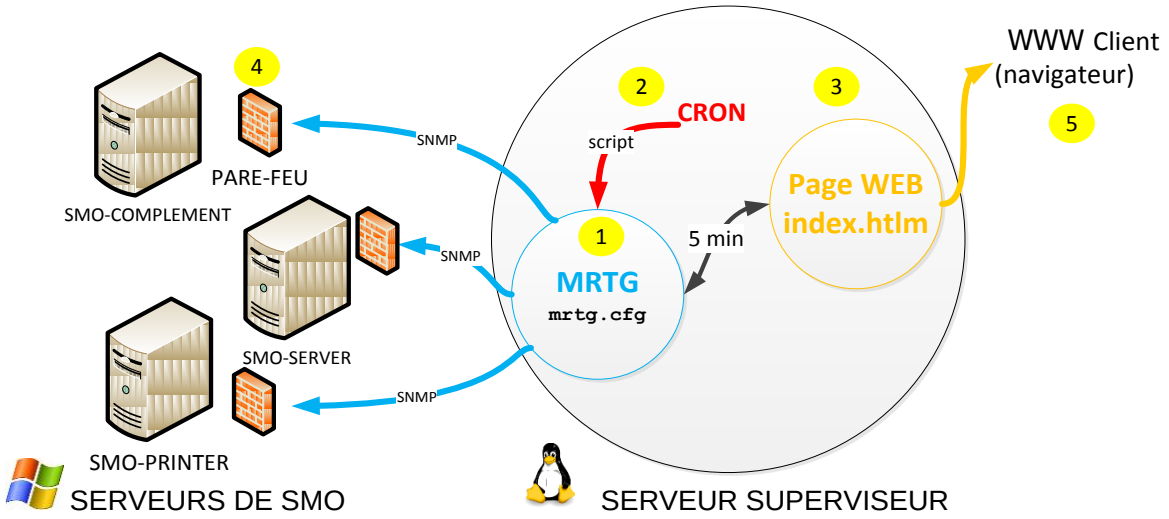


Figure 26 Fonctionnement de MRTG sur le serveur superviseur

La figure 26, montre les cinq étapes de la configuration de MRTG sur le serveur superviseur SMO. Ci-dessous le détaille des étapes pour la configuration de la mise en œuvre de MRTG. Notons que pour MRTG et Nagios l'installation et la configuration s'effectue en ligne de commande via l'éditeur VI[83]. Cet éditeur nous permet de configurer tous les composants.

Etape 1, nous commençons par installer le MRTG avec le gestionnaire de paquetage « Aptitude[84] » puis créons un répertoire « mrtg » et on attribue les droits d'accès à ce répertoire. Ce répertoire contient les données collectées via SNMP pour que MRTG puisse construire ses graphiques.

```
sudo aptitude install mrtg

sudo mkdir /var/www/mrtg

sudo chmod 755 /var/www/mrtg
```

La supervision des interfaces réseaux ETH0[85] des serveurs SMO est configurée avec « public[86] ».

[83] VI (Visual) est un éditeur de texte plein écran pour Linux.

[84] Aptitude est un gestionnaire de paquetage qui gère essentiellement les dépendances.

[85] ETH0 est la première interface réseau sur le serveur.

La commande « cfgmaker » ci-dessous crée un fichier mrtg.cfg contenant les informations des trois serveurs SMO avec comme paramètres les adresses IP des serveurs SMO [31].

Les paramètres utilisés sont les suivants :

« global 'WorkDir » est l'emplacement où sont enregistrés les fichiers html (graphiques).

« ifref=ip » permet d'atteindre l'interface (ETH0) des serveurs par leur adresse IP.

« global 'Options `[_]: bits` » désigne l'unité de mesure en bits.

« output=/ var.. » désigne l'emplacement du fichier de configuration de MRTG.

```
sudo cfgmaker –global 'WorkDir: /var/www/mrtg' –global 'Language: french' –ifref=ip –global'Options[_]: bits,growright' –output=/var/www/mrtg/mrtg.cfg public@192.168.169.1 public@192.168.169.5 public@192.168.169.8
```

Etape 2, paramétrage à 5 minutes de la tâche du CRON.

On crée une configuration pour mettre à jour la page Web du MRTG. Il faut pour cela, écrire le script « mrtgcron.sh » auquel nous attribuons le droit pour que la mise à jour des graphiques puisse s'exécuter toutes les 5 minutes via le « CRON[87] ».

```
#La configuration du fichier crontab exécute le script mrtgcron toutes les 5 minutes

crontab -l

*/5 * * * * /usr/local/bin/mrtgcron.sh
```

[86] Public est un système d'authenfication de SNMP non sécurisé qui permet d'envoyer via le datagramme UDP sur les ports 161 et 162 un « community name » mot de passe pour la lecture des OID.

[87] CRON permet l'exécution automatique de script sur les systèmes Unix, de commandes et de logiciels à une date et à une heure spécifiées à l'avance, ou selon un cycle défini à l'avance.

```
#fichier de configuration de mrtgcron.sh

#!/bin/sh

# /var/www/mrtg/

env LANG=C /usr/bin/mrtg /var/www/mrtg/mrtg.cfg
```

La commande « chmod » permet d'exécuter le script.

```
chmod a+x /usr/local/bin/mrtgcon.sh
```

Etape 3, création du fichier d'index pour la page Web.

La commande « indexmaker » génère les pages de MRTG [32].

```
Sudo indexmaker –tilte='MRTG server 192.168.169.1

–output=/var/www/mrtg/index.html /var/www/mrtg/mrtg.cfg
```

Etape 4, « iptables [88]» nous permet de paramétrer l'accès au serveur superviseur via le pare-feu [33].

« OUTPUT » trafic sortant du serveur superviseur sur la couche « UDP ».

La restriction dans le premier cas de figure ce fait sur le port de destination « dport 161 :162 ».

Dans le deuxième cas la restriction se fait sur le port source « sport 161 :162 »

```
iptables -A OUTPUT -p udp -s 192.168.xxx.xxx --sport 1024:65535

-d 0/0 --dport 161:162 -m state --state NEW,ESTABLISHED -j ACCEPT

iptables -A OUTPUT -p udp -s 192.168.xxx.xxx --sport 161:162 -d 0/0

--dport 1024:65535 -m state --state ESTABLISHED -j ACCEPT
```

« INPUT » trafic entrant sur le serveur superviseur.

[88] Iptables est l'interface en ligne de commande permettant de configurer le logiciel libre Netfilter sur Linux. Grâce à ce logiciel, l'administrateur système peut configurer les chaînes et règles dans le pare-feu.

La restriction dans le premier cas, la restriction se fait sur le port source “sport 161:162”.

Dans le deuxième cas, la restriction se fait sur le port destination “dport 161:162”

```
iptables -A INPUT -p udp -s 0/0 --sport 161:162 -d 192.168.xxx.xxx

--dport 1024:65535 -m state --state ESTABLISHED -j ACCEPT

iptables -A INPUT -p udp -s 0/0 --sport 1024:65535 -d 192.168.xxx.xxx
--dport 161:162 -m state --state NEW,ESTABLISHED -j ACCEPT
```

Etape 5, donner l’accès d’authentification par mot de passe à la page Web.

La modification du fichier « htaccess » consiste à attribuer un nom, un type et un fichier d’authentification.

```
#Extrait du fichier de configuration .htaccess

vi /var/www/html/mymrtg/.htaccess

AuthName "MRTG Graphs/Html restricted access"

AuthType Basic

AuthUserFile /var/members/.htpasswd
```

La commande “htpasswd” permet de générer un mot de passe avec le nom d’utilisateur “mrtgadmin”.

```
#Accès pour l'authentification par mot de passe à la page MRTG.

sudo htpasswd -c /var/members/.htpasswd mrtgadmin
```

A ce stade de la mise en place de la supervision, MRTG est opérationnel. Il nous reste à installer Nagios pour compléter la plate-forme de supervision.

6.5 Mise en œuvre de Nagios

Il faut maintenant installer et configurer l'ordonnanceur Nagios, les plugins, l'agent de transport NSClient++ et Postfix[89]. Rappelons que les pré-requis de Nagios ont été mis en place avec ceux de MRTG [30].

6.5.1 Installation de Nagios

Par sécurité on créer un compte utilisateur auquel sont attribués des droits. Les comptes et groupes servent à l'authentification des échanges entre les différents services tels que Nagios, le compilateur, Apache ou encore Postfix.

```
sudo /usr/sbin/useradd -m nagios -p nagios

sudo /usr/sbin/groupadd nagios

sudo /usr/sbin/usermod -G nagios nagios

sudo /usr/sbin/groupadd nagcmd

sudo /usr/sbin/usermod -a -G nagcmd nagios

sudo /usr/sbin/usermod -a -G nagcmd www-data
```

Il faut alors télécharger l'archive source, la désarchiver et la décompresser.

```
sudo wget ftp://xxx.xxx.xxx.xxx/nagios-3.2.1.tar.gz

tar xzf /install/nagios-3.2.1.tar.gz

tar xzf /install/nagios-plugins-1.4.14.tar.gz
```

A ce stade de l'installation, on accède au code source[90] de Nagios, accessible en lecture et compilation avec sa licence GPL. En Annexe 6 un extrait du code source « Nagios.C » .

---

[89] Postfix est un serveur de messagerie MTA (Mail Transfer Agent) que nous avons installé pour recevoir les alarmes par courriel.

[90] « Code source » est constitué d'instructions écrites dans un langage de programmation de haut niveau (langage C pour Nagios). Nagios étant programme sous licence GPL, permettent de lire, modifier et compiler le code source.

L'installation consiste à configurer Nagios avec les paramètres de sécurité liés au groupe « nagcmd », puis compiler le code source pour installer le serveur superviseur. On obtient alors des fichiers exécutables, des scripts[91] et les fichiers de configuration. Le script « configure » permet de configurer l'installation de Nagios avec les pré-requis (Postfix, Apache et Perl).

On se positionne dans le répertoire :

```
cd /install/nagios-3.2.1
```

On lance le script de configuration avant la compilation avec comme nom de groupe : « nagcmd »

```
./configure --with-command-group=nagcmd
```

Le résultat de la configuration permet de vérifier que Nagios et ses pré-requis ont bien été installés.

```
*** Configuration summary for nagios-3.2.1 ***:

 General Options:

 -------------------------

        Nagios executable:  nagios

        Nagios user/group:  nagios,nagios

       Command user/group:  nagios,nagcmd

            Embedded Perl:  yes

             Event Broker:  yes

        Install ${prefix}:  /usr/local/nagios

                Lock file:  ${prefix}/var/nagios.lock

  Check result directory:  ${prefix}/var/spool/checkresults

           Init directory:  /etc/init.d
```

[91] Les scripts de Nagios sont interprêtés principalement en Bash, Python, Perl.

```
 Apache conf.d directory:  /etc/apache2/conf.d

            Mail program:  /bin/mail

                 Host OS:  linux-gnu

Web Interface Options:

------------------------

                HTML URL: http://localhost/nagios/

                 CGI URL: http://localhost/nagios/cgi-bin/
```

On compile le noyau Nagios

```
sudo make all
```

On installe les programmes de base, CGIs, fichiers HTML[92]

```
sudo make install
```

On installe les scripts dans /etc/init.d

```
sudo make install-init
```

On installe et configure les commandes externes

```
sudo make install-commandmode

chmod g+s /usr/local/nagios/var/rw
```

On installe les fichiers de configuration par défaut dans /etc/local/nagios/etc

```
sudo make install-config
```

On installe la configuration Apache pour l'interface Web de Nagios.

```
sudo make install-webconf

sudo /usr/bin/install -c -m 644 sample-config/httpd.conf
/etc/apache2/conf.d/nagios.conf
```

[92] HTML Hyper Text Markup Language est un langage de balises pour écrire de l'Hypertext pour les pages Web.

L'authentification pour restreindre l'accès à Nagios, consiste à créer un nom d'utilisateur et un mot de passe pour l'accès aux CGI[93].

```
cp /usr/local/nagios/etc htpasswd -c htpasswd.users nagios

cp /usr/local/nagios/etc htpasswd -c htpasswd.users wwwrun

sudo htpasswd -c /usr/local/nagios/etc/htpasswd.users nagiosadmin
```

On redémarre le service Apache

```
sudo /etc/init.d/apache2 restart
```

A ce niveau de l'installation, Nagios est dépourvu de fonctionnalités, son noyau à besoin d'être complété par des plugins.

6.5.2 Installation des plugins

Les plugins sont des programmes ou scripts externes exécutés par Nagios à partir d'une ligne de commande pour contrôler un service ou un hôte local ou distant. Contrairement à beaucoup d'autres plateformes logicielles de supervision, Nagios ne possède pas de mécanisme qui lui permet de vérifier l'état d'un « host » ou d'un service. Nagios utilise le résultat de l'exécution des plugins pour déterminer l'état d'un hôte ou un service sur le réseau. Il existe de nombreux plugins standards pour Nagios qui peuvent être installés à partir de l'adresse suivante : nagiosplug.sourceforge.net [34].

Il est possible de créer ses propres plugins pour les adapter à des services spécifiques à l'entreprise. Le plugin peut être développé dans n'importe quel langage de programmation, il suffit de respecter la norme Nagios des codes retour ci-dessous :

- 0 OK (tout va bien)
- 1 WARNING (premier seuil d'alerte)
- 2 CRITICAL (deuxième seuil d'alerte)
- 3 UNKNOWN (impossible de connaître l'état du service)

[93] CGI (Common Gateway Interface) est un standard industriel qui indique comment envoyer une requête aux serveurs http.

Nagios utilise ce message lors de l'envoi d'une notification. Les règles de développement de Nagios figurent à l'adresse suivante : nagiosplug.sourceforge.net/developer-guidelines.html [35].

Dans la configuration Nagios, les plugins s'exécutent soit en « mode local », soit en « mode distant ». La figure 27 ci-dessous illustre le « mode local ».

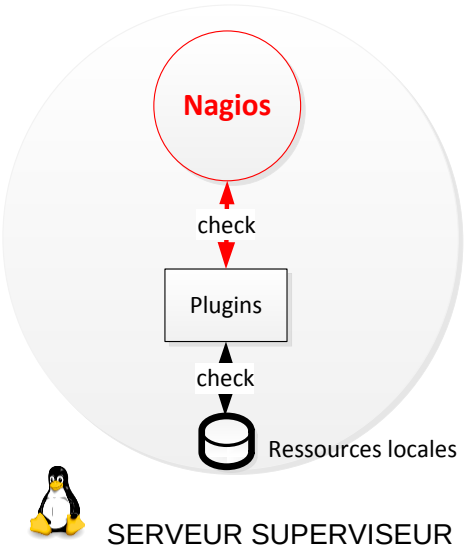


Figure 27 Contrôle des ressources en mode local

La figure 28 ci-dessous illustre le « mode distant », par exemple pour la vérification de la commande Ping sur les serveurs de SMO. La commande « check_ping[94] » permet d'envoyer une requête CMIP (de contrôle et d'erreur), sur une autre machine.

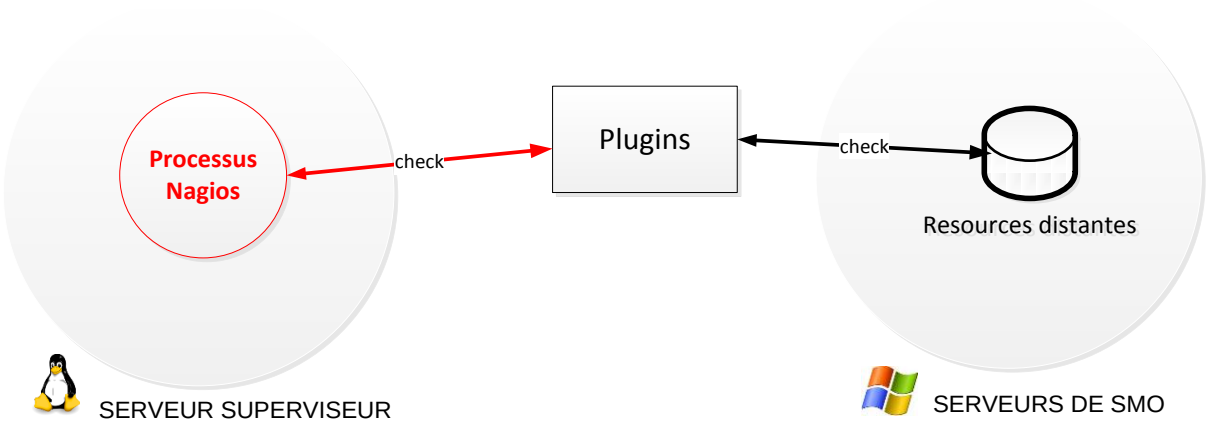


Figure 28 Contrôle des ressources en mode distant

[94] En annexe 8 un extrait du plugin « Nagios Check_ping ».

Pour l'installation des plugins, on procède comme pour la partie précédente, à la configuration, la compilation et l'installation des plugins V1.4.1.14, avec comme paramètres de configuration le protocole SSL[95], le « Groupe » et « l'Utilisateur ».

On décompresse les plugins

```
sudo wget ftp://xxx.xxx.xxx.xxx/nagios-plugins-1.4.14.tar.gz

sudo tar xzf /install/nagios-plugins-1.4.14.tar.gz

cd nagios-plugins-1.4.14
```

On configure les paramètres avant compilation

```
sudo ./configure --with-nagios-user=nagios --with-nagios-group=nagios

--with-openssl
```

On compile

```
sudo make
```

On installe

```
sudo make install
```

Configuration du démon[96] Nagios afin qu'il soit lancé au démarrage de Linux.

```
sudo ln -s /etc/init.d/nagios /etc/rcS.d/S99nagios
```

Une fois les plugins installés, Nagios a besoin d'un agent de transport pour contrôler les composants des serveurs (mémoire, CPU, disque dur, uptime, explorer).

6.5.3 Les agents de transport

En s'inspirant de la structure de gestion de réseaux (MNS) définie par la norme ISO 7498-4, les développeurs de Nagios ont conçu différents agents de transport pour effectuer des vérifications sur une ressource distante sans avoir à modifier la

[95] SSL (Secure Socket Layer) est un protocole de communication sécurisé utilisé pour l'authentification la confidentialité (chiffrement) et l'intégrité.

[96] Démon est un processus qui s'exécute en arrière plan.

politique de sécurité mise en place. Figure 29, une vue de l'agent de transport représenté sur les serveurs SMO.

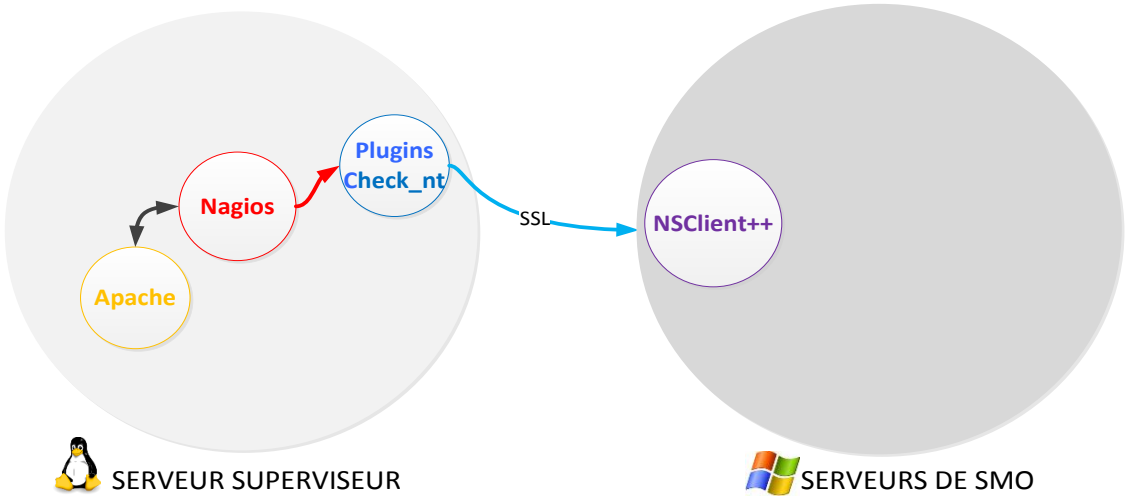


Figure 29 Agent de transport NSClient++

Les deux principaux agents de transport sont NRPE (Nagios Remote Plugin Executor) et NSCA (Nagios Service Check Acceptor). Pour la configuration de SMO nous avons installé le plugin NSClient++ [38] qui intègre les deux agents NRPE et NSCA, représentés dans la figure 30 ci-dessous. L'agent NRPE utilise un mode actif, la demande d'exécution des plugins est faite à l'initiative du serveur Nagios. L'agent NSCA utilise un mode passif, c'est à l'initiative du démon NSClient++ que s'exécute la requête.

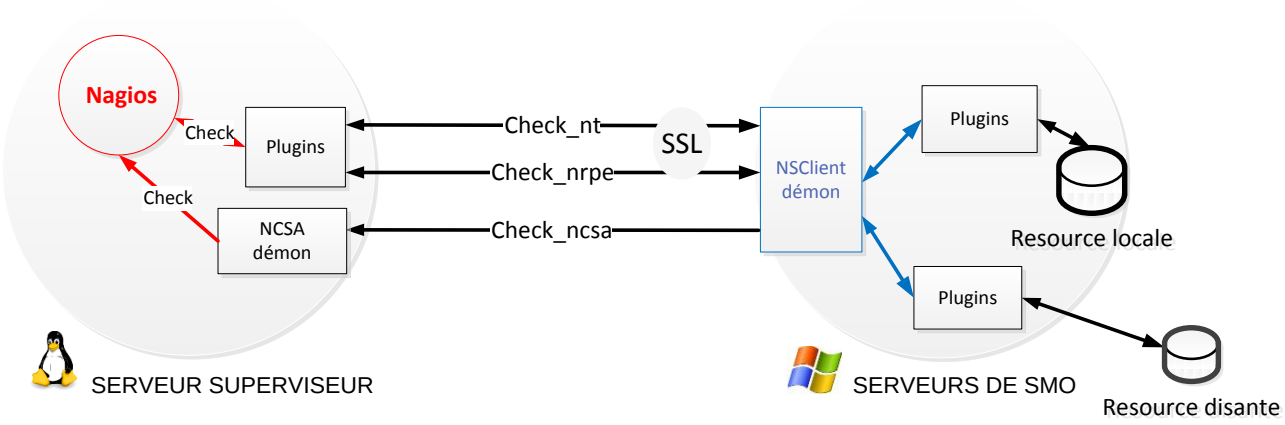


Figure 30 Agents de transport NRPE et NSCA

NRPE est le moyen de transport pour effectuer des contrôles au niveau local. En effet, avec NRPE la demande d'exécution des plugins est faite à l'initiative du serveur de supervision. Le plugin « check_nrpe » permet à Nagios d'envoyer sur une machine distante, les instructions au démon NRPE [36]. Ce mode comporte de nombreux avantages tels que l'encryptage des données pendant leur transport.

NSCA est la méthode dite passive. NSCA est installé sur chaque hôte et service distant. En exécutant les plugins passifs sur les machines à surveiller, NSCA à l'avantage de pallier les problèmes liés aux réseaux sécurisé par pare-feu ou les processus nécessitant une fréquence d'exécution très courte [37].

6.5.4 Installation de NSClient++

A partir des serveurs Windows NT SMO, il faut télécharger l'agent NSClient++ à la source serveur « nsclient.org » puis le décompresser dans le répertoire c:\program files\nsclient++\

L'installation de NSClient++ est exécutée en ligne de commande (CMD).

```
c:\program files\nsclient++\nsclient++ /install
```

La configuration de l'agent NSClient++, se fait au travers du fichier NSC.ini dans le répertoire « c:\program flies\nsclient++ ».

Cette configuration permet de contrôler le CPU, la mémoire, la partition système, l'Uptime et l'explorer des trois serveurs SMO, figure 31.

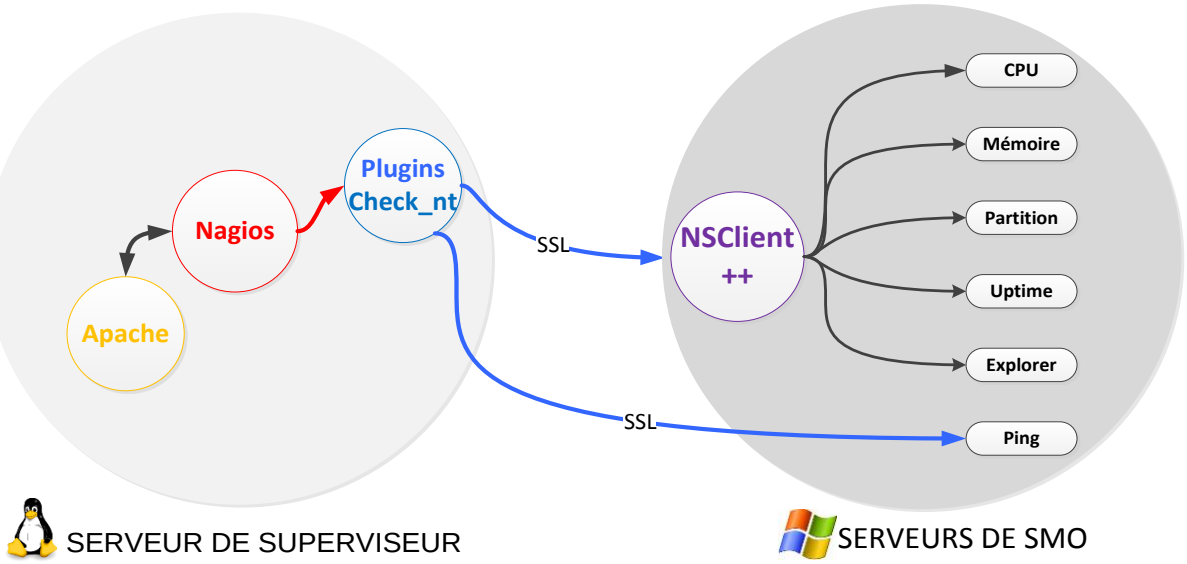


Figure 31 Agent de transport NSClient++

- Uptime : permet de contrôler le temps d'activité sans interruption du serveur.
- RAM (Random Access Memory): mémoire vive
- CPU : Central Processeur Unit
- Partition système : mémoire de masse
- Explorer : permet l'affichage des dernières versions des systèmes d'exploitation windows.

On configure NSClient++ sur les serveurs SMO. Ci-dessous un extrait du fichier de configuration «nsclient++.ini ».

```
#Extrait du fichier Nsclient++.ini

[modules]
CheckDisk.dll
NSClientListener.dll
NRPEListener.dll
Checksystem.dll
NSCAgent.dll
[Settings]
allowed_hosts=127.0.0.1/32,192.168.xxx.x
use_file=1
[log]
debug=1
file=nsclient.log
[NSClient]
allowed_hosts=192.168.xxx.x
port=12489
```

Le module « CheckDisk.dll[97] » permet de contrôler un disque dur et la taille du volume de la partition ainsi que les répertoires et fichiers.

Le module « Checksystem.dll » permet de vérifier la charge du CPU, l'Uptime, l'état d'un service, l'état d'un processus, l'usage de la mémoire.

Le module « NSClientListener.dll » autorise l'utilisation de la commande « check_nt [98]» avec ses propres variables :

[97] dll est une librairie logicielle de programmes réutilisables.

- « check_nt!USEDDISKSPACE »,
- « check_nt!MEMUSE »,
- « check_nt!UPTIME »,
- « check_nt!CPULOAD »,
- « check_nt!USEDDISKSPACE »,
- « check_nt!PROCSTATE ».

Le module « NRPEListener.dll» permet l'utilisation de la commande « chech_nrpe ».

Le module « NSCAgent.dll » soumet les notifications au serveur en mode passif.

A ce stade, la configuration des serveurs Windows SMO est terminée. Il faut alors vérifier que la configuration fonctionne en exécutant les commandes suivantes sur le serveur superviseur Nagios.

La commande suivante vérifie le fonctionnement de l'agent NSClient++. L'agent de transport retourne son numéro de version installé sur les serveurs SMO. Les paramètres de la commande check_nt sont, « -H » l'adresse IP, « -p » le numéro de port, «-v » la variable CLIENTVERSION de NSClient++.

```
sudo /usr/local/nagios/libexec$ ./check_nt -H 10.1.240.11 -p 12489 -v

CLIENTVERSION

NSClient++ 0.3.7.493 2009-10-12
```

A titre d'exemple, la commande suivante retourne le volume de la partition d:\. La variable USEDDISKSPACE à comme paramètre « -w » (warning) à plus de 80%, « -c » (critical) à plus de 90% du volume de la partition.

```
sudo /usr/local/nagios/libexec$ ./check_nt -H 10.1.240.11 -p 12489 -v
USEDDISKSPACE -l d -w 80 -c 90

d:\ - total: 5363.86 Gb - used: 5312.61 Gb (99%) - free 51.25 Gb (1%)
| 'd:\ Used Space'=5312.61Gb;4291.09;4827.47;0.00;5363.86
```

[98] En Annexe 7 un extrait du plugin « Nagios check_nt ».

Le chapitre suivant traite de la configuration de Nagios. Notons que cette tâche peut se faire avant l'installation des plugins.

6.5.5 Configuration des composants

Une fois Nagios compilé les fichiers « .cfg » doivent être configuré à l'environnement de SMO. Cette configuration est complexe ; pour en faciliter la compréhension, seuls les principaux fichiers de configuration ont été représentés. La figure 32 ci-dessous est donc un résumé des connexions existantes entre les différents composants configurés. Afin d'illustrer les différentes étapes de cette configuration nous donnons une définition pour chacun des éléments de la figure 32 ainsi que les extraits des fichiers de configuration (Contacts, services, Timperiods) [28].

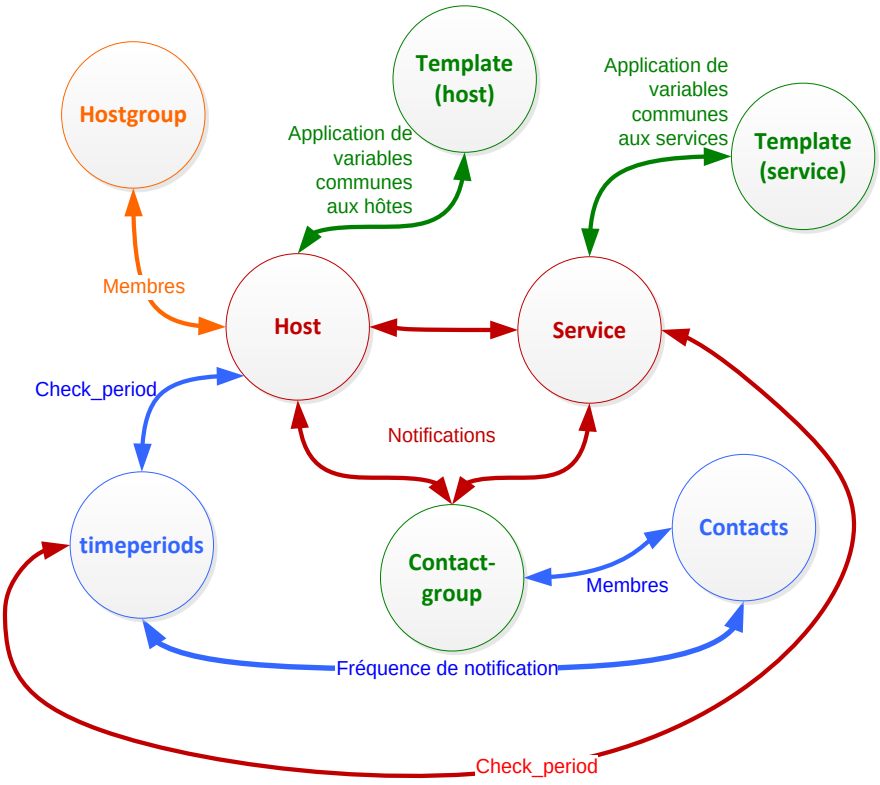


Figure 32 Principaux composants de la configuration

Les fichiers de configuration sont répartis selon le type d'infrastructure, serveur linux, serveur Windows, composant réseaux, imprimantes. Cette répartition permet de ne modifier qu'un seul fichier lors d'un ajout de serveur ou de composant réseau.

- « hosts » (hôtes) représentent tous les composants du LAN de SMO.

- « hostgroups » sont des groupements de « hosts » fait à partir de critères communs (rôle, application). Pour la configuration de SMO, nous avons définis trois groupes Linux Serveur (serveur superviseur), Windows serveur (Serveurs SMO) et Network serveur (pare-feu).

- « Templates » (modèles d'hôtes et services) permettent de limiter les définitions de services et d'hôtes en groupant les variables communes. Chaque création d'un nouvel élément peut hériter de la configuration d'un modèle « Template ». En annexe 00 le fichier de configuration « templates.cfg » des modèles que nous avons configurés.

- « Contacts » sont les collaborateurs alertés par la Nagios. Pour SMO nous avons choisi d'alerter un « contact », le responsable TI (Technologies de l'Information) par email. Il est possible d'ajouter d'autres moyens de notifications tels que le sms. L'extrait du fichier « contacts.cfg » ci-dessous montre le «contact » et les paramètres configurés des notifications des « hosts » et « services ».

Les paramètres du « service_notification_options » sont :

- w (warning), le service en état avertissement
- u (unknown), le service est en état inconnu
- c (critical), le service est en état critique
- r (recovery), le service est revenu dans en état OK

Les paramètres du « host_notification_options » sont :

- d (down) l'hôte est éteint
- r (recovery) l'hôte est de nouveau accessible

```
 #Extrait du fichier de configuration contacts.cfg
define contact{
       contact_name                    SUSS-Windows            ;
Short name of user
       use                             generic-contact         ;
```

```
Inherit default values from generic-contact template (defined above)
       alias                           SUSS Admin              ; Full
name of user
       service_notification_period     24x7
       host_notification_period        24x7


       service_notification_options    w,u,c,r

       host_notification_options       d,r


       service_notification_commands   notify-service-by-email

       host_notification_commands      notify-host-by-email

       email                           suss.nagios@gmail.com   ;
       }
```

« Services » sont les services configurés :

- « check_ping »,
- « check_nt!USEDDISKSPACE »,
- « check_nt!MEMUSE »,
- « check_nt!UPTIME »,
- « check_nt!CPULOAD »,
- « check_nt!USEDDISKSPACE »,
- « check_nt!PROCSTATE ».

Les deux extraits suivants concernent la configuration des services pour les serveurs SMO.

Dans le premier exemple, nous avons un service de Ping avec comme paramètres le modèle, le serveur, le service, le « contact_groupe » et les paramètres de la commande « check_command » (si le temps de réponse > 200 ms ou la perte de paquets > 20% envoi d'une alerte WARNING. Si le temps de réponse > 800ms ou la perte de paquets > 60% envoi de l'alerte CRITICAL).

```
#Extrait du fichier de configuration services.cfg
define service{
       use                     generic-service
```

```
        host_name               SMO-Server
        service_description     ping
        check_command           check_ping!200.0,20%!800.0,60%
        contact_groups          SUSS-Win


        }
```

Dans l'exemple suivant, nous utilisons le plugin « check_nt " avec la variable MEMUSE qui interroge l'agent NsClient++ pour vérifier l'usage de la mémoire. Si le paramètre fixé à 80% d'utilisation de la mémoire est atteint, une notification d'alerte « warning » est envoyée au contact, à 90% la notification est « critical ».

```
#Extrait du fichier de configuration services.cfg
define service{
        use                     generic-service

        host_name               SMO-Printers
        service_description     Memory Usage
        check_command           check_nt!MEMUSE!-w 80 -c 90
        }
```

- Timeperiods, (périodes de temps) servent à fixer les plages de contrôles des « services » des « hosts » et les notifications aux « contacts ».

L'extrait du fichier « Timeperiods » ci-dessous, montre que les notifications et les contrôles se font 24 heures sur 24, 7 jours sur 7.

```
#Extrait du fichier de configuration timeperiodes.cfg
Define timeperiod{
       timeperiod_name 24x7
       alias           24 Hours A Day, 7 Days A Week
       sunday          00:00-24:00
       monday          00:00-24:00
       tuesday         00:00-24:00
       wednesday       00:00-24:00
       thursday        00:00-24:00
       friday          00:00-24:00
       saturday        00:00-24:00
       }
```

Les notifications d'alertes présentées dans les extraits de fichiers ci-dessus nécessitent la mise en place du serveur de messagerie MTA (Mail Transfer Agent) Postfix pour délivrer les alertes. Dans le point suivant, nous traitons du mécanisme de notification et de Postfix.

6.5.6 Alarmes

Si la représentation graphique de l'état des ressources en temps réel est primordiale pour la supervision du réseau elle ne peut se passer des alertes, autre fonctionnalité importante de Nagios [28]. Grâce à son interfaçage avec un serveur SMTP[99], Nagios peut prévenir les administrateurs dès qu'un événement ou incident survient grâce à un mécanisme de notifications qui utilise le MTA (Mail Transfer Agent) Postfix.

[99] SMTP (Simple Mail Transfer Protocol) est un protocole de communication pour les courriels vers les serveurs de messagerie électronique. Ce protocole n'authentifie pas l'expéditeur, il faut le compléter par l'extension SMTP-AUTH

Notifications

La figure 33 ci-dessous décrit le déroulement de la notification d'alerte de Nagios.

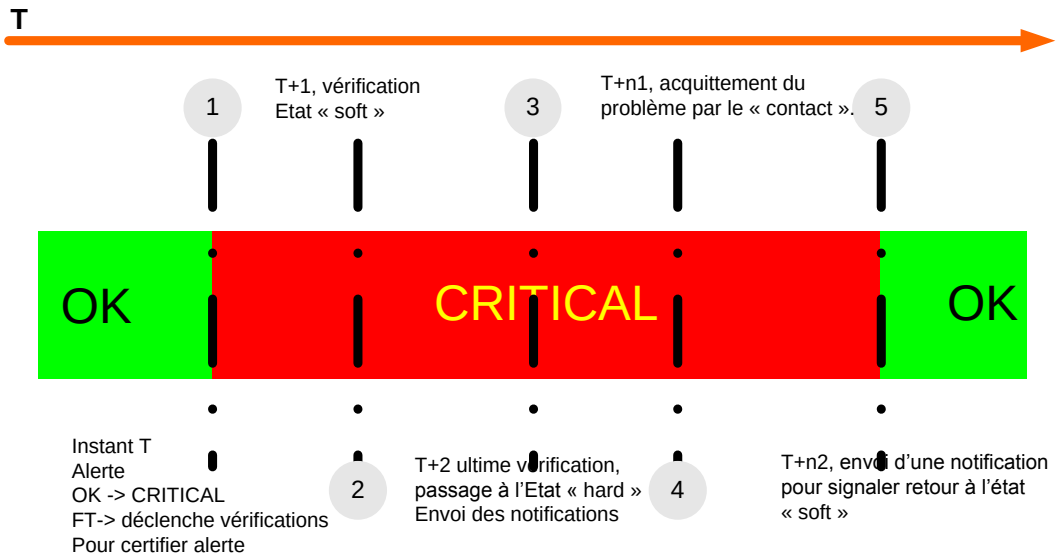


Figure 33 Processus de la notification

A l'instant T, « hosts » et « services » supervisés changent d'état, ils passent de « ok » à « critique ». Cette étape enclenche un cycle de vérification des incidents.

T+1, la première vérification.

T+2, fin du cycle de vérification, Nagios « certifie » l'incident et bascule « hosts » et « services » à l'état «hard ». Le cycle de notifications peut démarrer.

T+n1, les notifications sont stoppées après avoir été acquittées par le « contact ».

Il est à noter que « l'acquittement » des notifications est un des points importants du SMSI et est souvent contrôlé lors des audits. Cette action marque implicitement la prise en charge du problème par le « contact ».

L'envoi des notifications au « contact » s'effectue par le biais de MTA (Mail Transfert Agent) Postfix.

MTA (Mail Transfer Agent) Postfix

Postfix est un serveur de messagerie (MTA) sous licence libre (IBM Public Licence) utilisé par défaut sur les systèmes Unix, Linux et Mac OS X.

Par rapport à ses prédécesseurs tel que Sendmail[100], Postfix utilise plusieurs niveaux de protection afin de prévenir le système de toute intrusion. Il n'existe par exemple pas de lien direct entre les programmes sensibles tels que la livraison des courriels locaux et le réseau. Les messages d'alertes sont envoyés à partir d'un MTA Postfix vers un autre MTA (destinataire) en utilisant le protocole SMTP. Ce dernier transfère le message au MDA[101] (Mail Delivery Agent) l'agent de livraison qui gère le stockage des messages jusqu'à ce que le MUA[102] (Mail User Agent) vienne les récupérer via un protocole POP ou IMAP [40].

La figure 34 montre le parcours d'une alerte par courrier électronique du serveur superviseur à la boite aux lettres de l'administrateur.

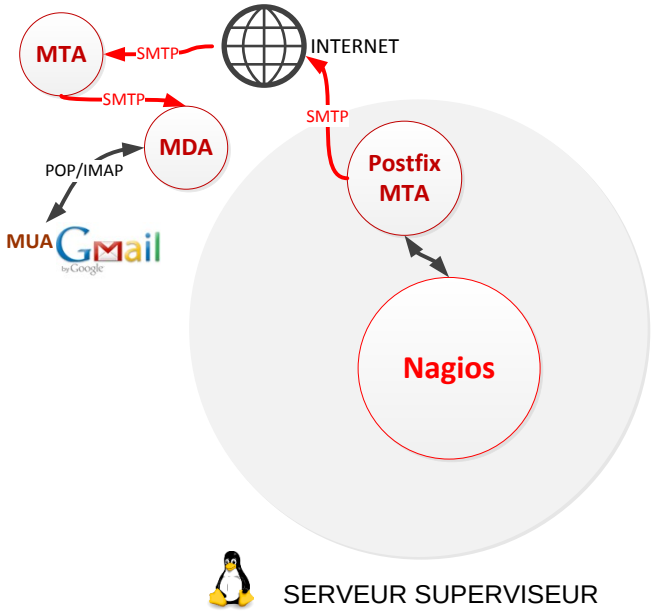


Figure 34 MTA Postfix

Postfix s'installe en quatre étapes comme les paquetages précédents, installation, configuration, redémarrage et vérification.

---

[100] Sendmail, avec un code source ouvert, est le plus populaire MTA sur internet. En 2007 une étude estimait que presque un tiers des serveurs de messagerie accessibles publiquement utilisaient Sendmail.

[101] MDA (Mail Delivery Agent) est le logiciel final du voyage d'un courrier électronique avant d'être livré à l'utilisateur sur le MUA.

[102] Le MUA (Mail User Agent) est dans notre cas, l'interface Gmail.

Nous sommes arrivés au terme de l'installation de la plate-forme logicielle de supervision, notre serveur superviseur est opérationnel.

A travers l'interface Web, nous allons pouvoir consulter les premiers enregistrements effectués par Nagios et MRTG via SNMP sur les « services » et « Hosts » de SMO.

Le choix des éléments c'est-à-dire les actifs comprenant les « services » et les « hosts » ainsi que le périmètre à superviser, sont des étapes essentielles à la mise en œuvre de Nagios. La méthode PDCA de la norme ISO/CEI 27001 nous a été d'une grande utilité pour planifier le travail et servira à faire évoluer la plate-forme de supervision.

Dans la partie qui suit nous livrons un aperçu de l'utilisation de Nagios et de MRTG et donnons quelques exemples extraits des enregistrements de MRTG avec lesquels nous montrons comment exploiter ces vues pour vérifier les indicateurs du SMSI de SMO.

7. Enregistrements de Nagios et de MRTG

7.1 Utilisation de Nagios

La connexion à l'interface de Nagios se fait par le biais d'un navigateur Web à partir de n'importe quel poste du LAN SMO après authentification par nom d'utilisateur et mot de passe.

- Adresse sur le LAN : http://192.168.xxx.x/nagios

La page de navigation propose de nombreuses vues des différents enregistrements. Les vues sont réparties sur deux principaux menus, « current status » supervision et « reports » enregistrements/rapports résumés ci-dessous.

7.1.1 Vues du « monitoring »

« Tactical overview » figure 35, est une vue générale sur laquelle se situent les principales informations telles que les rapports de statistiques et les notifications des services et des « hosts ». La « tactical overview » permet de naviguer facilement et d'accéder directement à l'information.

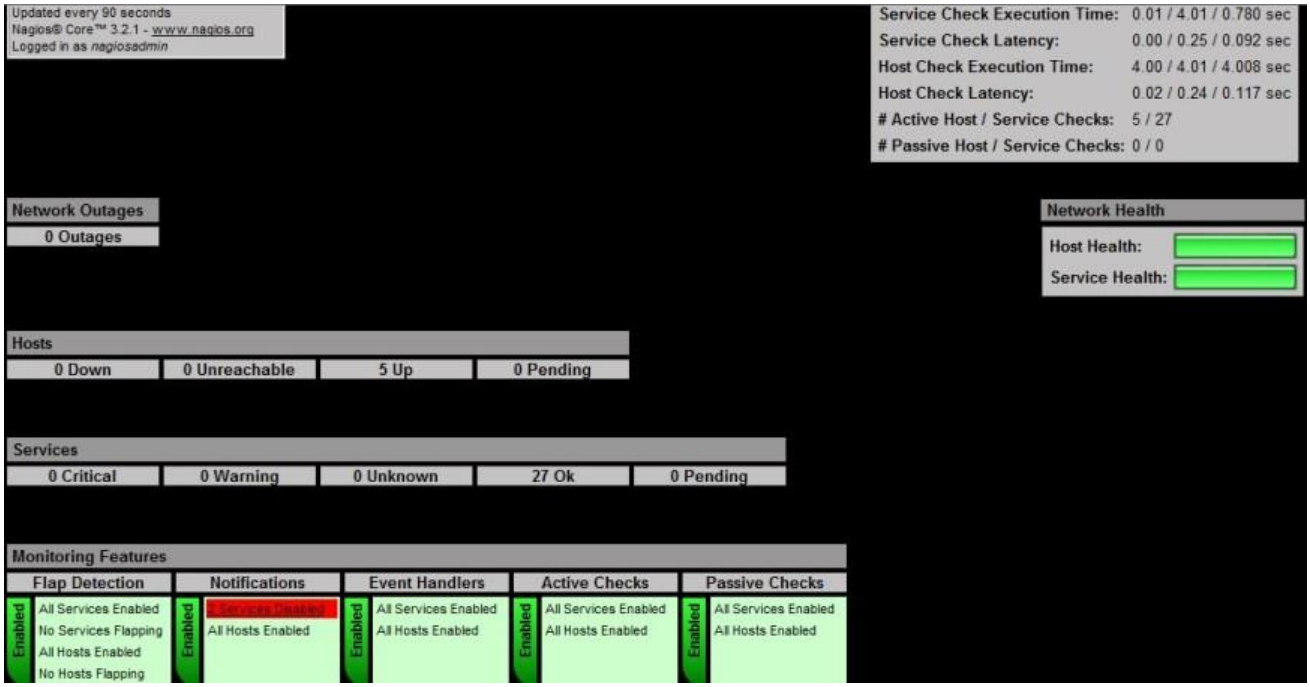


Figure 35 "Tactical overview" de Nagios

« Map » figure 36, est une carte de l'état des « Hosts » dotée d'infobulles[103]. Les relations de dépendances, les flux et la topologie des « hosts » peuvent aussi y être représentés.

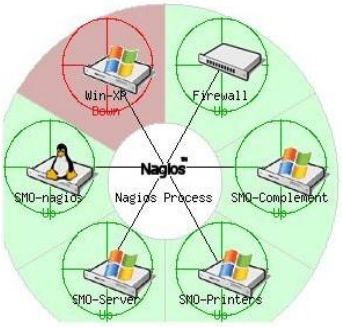


Figure 36 "Map" des "hosts" du SMSI supervisé

[103] Infobulle est un message qui apparaît lors du passage du curseur de la souris sur certains éléments.

« Service detail» donne une liste exhaustive des services supervisés par « hosts ». La vue comprend le statut de l'alerte « Ok, Critical, Warning, Unknown », la date et l'heure de la dernière vérification, le temps écoulé depuis la dernière notification, le nombre de vérifications et le statut de chaque événement. La figure 37 montre le « service detail » des différents services contrôlé sur le serveur SMO-Complement.

Host	Service	Status	Last Check	Duration	Attempt	Status Information
Firewall	PING	OK	05-28-2010 16:32:49	0d 6h 57m 3s	1/3	PING OK - Paquets perdus = 0%, RTA = 0.55 ms
SMO-Complement	C:\ Drive Space	OK	05-28-2010 16:31:16	0d 3h 3m 36s	1/3	c: - total: 48.83 Gb - utilisÃ© 9.52 Gb (19%) - libre 39.31 Gb (81%)
	CPU Load	OK	05-28-2010 16:34:24	0d 2h 50m 28s	1/3	Charge CPU 1% (5 moyenne minimale)
	Explorer	OK	05-28-2010 16:25:07	0d 2h 49m 45s	1/3	Explorer.EXE: Running
	Memory Usage	OK	05-28-2010 16:28:53	0d 2h 47m 47s	1/3	MÃ©moire utilisÃ©e: total:3944.01 Mb - utilisÃ©e: 1740.50 Mb (44%) - libre: 2203.51 Mb (56%)
	Uptime	OK	05-28-2010 16:27:12	0d 2h 47m 40s	1/3	SystÃ¨me dÃ©marrÃ© - 18 jour(s) 6 heure(s) 27 minute(s)
	ping	OK	05-28-2010 16:28:40	0d 7h 6m 12s	1/3	PING OK - Paquets perdus = 0%, RTA = 0.16 ms

Figure 37 "Service detail" du serveur SMO-Complement

« Hosts » figure 38, donnent une vue des groupes des éléments contrôlés qui ont été définis dans les fichiers de configuration.

Figure 38 "Hosts" sur Nagios

7.1.2 Vues du « reporting »

Le « reporting » est un service de Nagios qui fournit un historique des résultats d'exécution des plugins et un rapport de disponibilité des ressources (hosts, services) présentés sous différents angles.

« Availability » offre une vue sous forme de tableau de la disponibilité des « hosts » et services. La disponibilité est fonction du temps de bon fonctionnement d'un « host » et de tous ses services.

Dans la figure 39 on constate que la disponibilité des serveurs est à 100% alors que le pare-feu qui est resté inactif 3j /8h /29m montre un « Time Up » de seulement 52%.

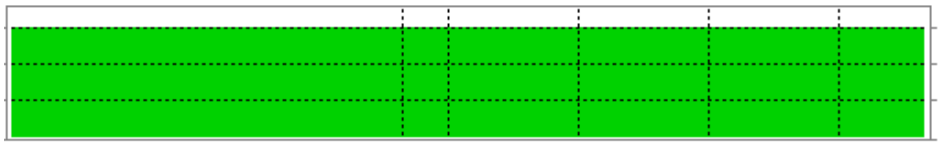
State Breakdowns:

Up : (52.087%) 3d 15h 30m 21s
Down : (0.000%) 0d 0h 0m 0s
Unreachable : (0.000%) 0d 0h 0m 0s
Indeterminate: (47.913%) 3d 8h 29m 39s

Host	% Time Up	% Time Down	% Time Unreachable	% Time Undetermined
Firewall	52.072% (100.000%)	0.000% (0.000%)	0.000% (0.000%)	47.928%
SMO-Complement	100.000% (100.000%)	0.000% (0.000%)	0.000% (0.000%)	0.000%
SMO-Printers	100.000% (100.000%)	0.000% (0.000%)	0.000% (0.000%)	0.000%
SMO-Server	100.000% (100.000%)	0.000% (0.000%)	0.000% (0.000%)	0.000%
SMO-nagios	100.000% (100.000%)	0.000% (0.000%)	0.000% (0.000%)	0.000%
Average	90.414% (100.000%)	0.000% (0.000%)	0.000% (0.000%)	9.586%

Figure 39 Rapport de disponibilité des "hosts"

« Trends » cette vue présenté dans la figure 40, affiche un historique des tendances (indisponible, indéterminé, non fonctionnel, fonctionnel) des « hosts » et « services » supervisés.



Up : (100.000%) 7d 0h 0m 0s
Down : (0.000%) 0d 0h 0m 0s
Unreachable : (0.000%) 0d 0h 0m 0s
Indeterminate: (0.000%) 0d 0h 0m 0s

Figure 40 "Trends" du serveur SMO

« Alerts » ce point propose différentes représentations graphiques des alertes à commencer par un « historic » figure 41, qui affiche une liste par date et par heure de toutes les alertes et retours à la normale. Cette liste comporte également la date et l'heure des éventuelles modifications de la configuration (redémarrage de la configuration) et de Nagios (démarrage, arrêt, redémarrage du démon).

July 30, 2010 15:00

[07-30-2010 15:30:03] SERVICE ALERT: SMO-Server;ping;OK;SOFT;2;PING OK - Packet loss = 0%, RTA = 0.18 ms
[07-30-2010 15:28:03] SERVICE ALERT: SMO-Server;ping;WARNING;SOFT;1;PING WARNING - DUPLICATES FOUND! Packet loss = 0%, RTA = 17.40 ms

July 30, 2010 12:00

[07-30-2010 12:48:03] SERVICE ALERT: SMO-Server;ping;OK;SOFT;2;PING OK - Packet loss = 0%, RTA = 0.43 ms
[07-30-2010 12:46:03] SERVICE ALERT: SMO-Server;ping;WARNING;SOFT;1;PING WARNING - DUPLICATES FOUND! Packet loss = 0%, RTA = 30.42 ms

Figure 41 "alerts history"

« histogram » figure 42 est une autre vue des alertes qui permet d'observer sur un graphique l'historique des événements survenus sur un mois.

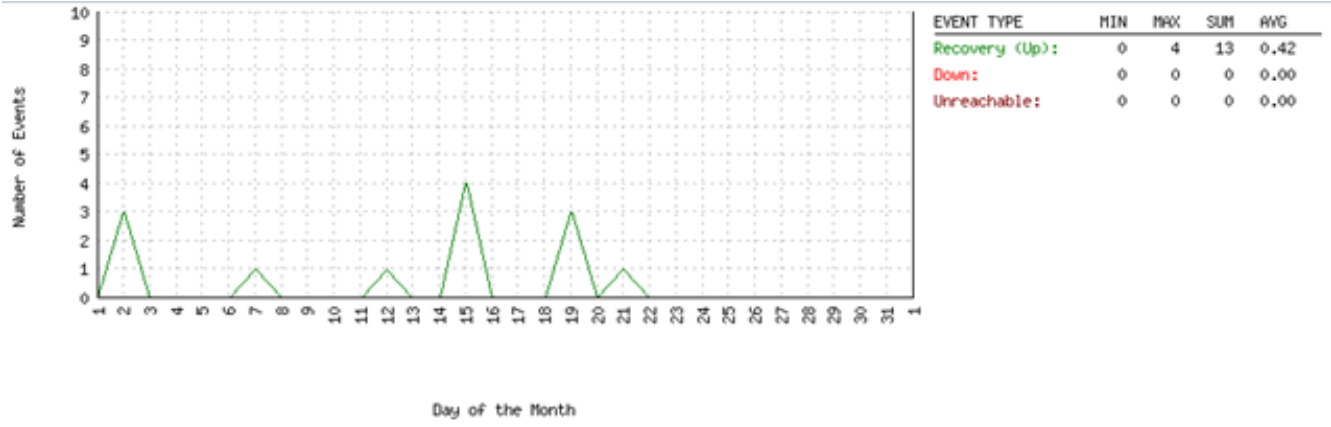


Figure 42 "histogram" des alertes

« Summary » figure 43 cette vue présente sous forme de tableau, un historique de l'état et du retour des alertes.

Displaying most recent 25 of 36 total matching alerts

Time	Alert Type	Host	Service	State	State Type	Information
07-30-2010 15:30:03	Service Alert	SMO-Server	ping	OK	SOFT	PING OK - Packet loss = 0%, RTA = 0.18 ms
07-30-2010 15:28:03	Service Alert	SMO-Server	ping	WARNING	SOFT	PING WARNING - DUPLICATES FOUND! Packet loss = 0%, RTA = 17.40 ms
07-30-2010 12:48:03	Service Alert	SMO-Server	ping	OK	SOFT	PING OK - Packet loss = 0%, RTA = 0.43 ms
07-30-2010 12:46:03	Service Alert	SMO-Server	ping	WARNING	SOFT	PING WARNING - DUPLICATES FOUND! Packet loss = 0%, RTA = 30.42 ms
07-30-2010 07:06:03	Service Alert	SMO-Server	ping	OK	SOFT	PING OK - Packet loss = 0%, RTA = 0.18 ms
07-30-2010 07:04:03	Service Alert	SMO-Server	ping	WARNING	SOFT	PING WARNING - DUPLICATES FOUND! Packet loss = 0%, RTA = 17.49 ms

Figure 43 "Summary" des alertes

« Notifications », cette vue donne un rapport des dernières notifications, escalades et acquittements des « services » et « hosts ». Elle se compose principalement du groupe de « contact » et de la «notification command » (courriel) ainsi que du statut de chaque événement. Dans le cas présenté en figure 44, aucune remontée d'alerte n'a été enregistrée.

Figure 44 "Notifications" des alertes

Les copies d'écran « Services status detail » et « Notifications » affichent les informations des alertes acquittées et non acquittées. Dans la figure 45 on remarque que l'état des différents « services » du pare-feu et du serveur « SMO-Complement » sont tous « OK ».

Host	Service	Status	Last Check	Duration	Attempt	Status Information
Firewall	PING	OK	05-28-2010 16:32:49	0d 6h 57m 3s	1/3	PING OK - Paquets perdus = 0%, RTA = 0.55 ms
SMO-Complement	C:\ Drive Space	OK	05-28-2010 16:31:16	0d 3h 3m 36s	1/3	c: - total: 48.83 Gb - utilisÃ© 9.52 Gb (19%) - libre 39.31 Gb (81%)
	CPU Load	OK	05-28-2010 16:34:24	0d 2h 50m 28s	1/3	Charge CPU 1% (5 moyenne minimale)
	Explorer	OK	05-28-2010 16:25:07	0d 2h 49m 45s	1/3	Explorer.EXE: Running
	Memory Usage	OK	05-28-2010 16:28:53	0d 2h 47m 47s	1/3	MÃ©moire utilisÃ©e: total:3944.01 Mb - utilisÃ©e: 1740.50 Mb (44%) - libre: 2203.51 Mb (56%)
	Uptime	OK	05-28-2010 16:27:12	0d 2h 47m 40s	1/3	SystÃ¨me dÃ©marrÃ© - 18 jour(s) 6 heure(s) 27 minute(s)
	ping	OK	05-28-2010 16:28:40	0d 7h 6m 12s	1/3	PING OK - Paquets perdus = 0%, RTA = 0.16 ms

Figure 45 Rapport sur les "service status details" du serveur SMO-Printer

En résumé, on constate que l'utilisation de Nagios est simple et offre de nombreuses vues de l'état des « services » et « hosts ». A travers le « monitoring » et le « reporting », Nagios livre une analyse approfondie de l'état du système et de précieuses informations sur les besoins du réseau. L'outil MRTG, vient renforcer la supervision de Nagios avec de multiples graphiques de métrologie extraits du trafic réseau sur les équipements.

7.2 Utilisation de MRTG

La connexion au service MRTG s'effectue comme pour Nagios, par le biais d'un navigateur Web et de l'authentification de l'utilisateur à l'adresse suivante :

- http://192.168.xxx.x/mrtg

L'outil de supervision MRTG propose une multitude de graphique qui nous donne l'état en temps réel de tous les équipements supervisés du réseau. Dans les figures ci-dessous nous présentons quelques exemples qui illustrent les possibilités de cet outil.

La figure 46, montre une perte du réseau d'environ vingt minutes à 10h30 pouvant être causé par un « conflit IP[104] ».

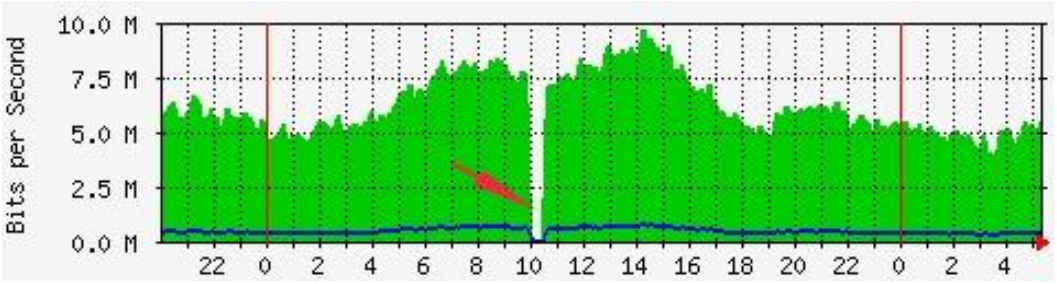


Figure 46 Perte de trafic sur le réseau

Dans la figure 47, le graphique révèle un pic de la bande passante sur trois heures.

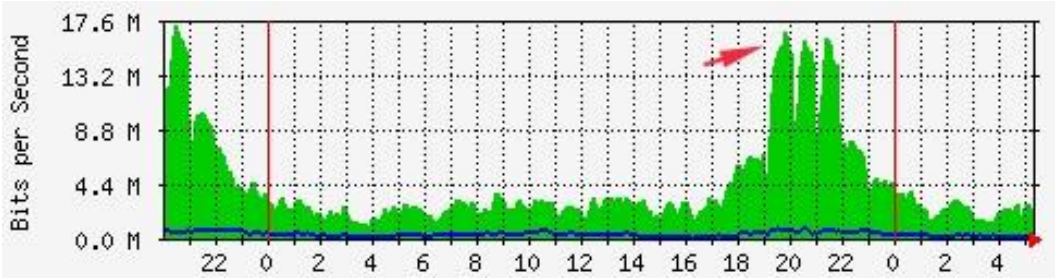


Figure 47 Pic de la bande passante sur le réseau

[104] Conflit IP lorsque deux serveurs, imprimantes ou postes ont une adresse IP identique sur le même réseau, une de ces machines s'arrête. Il faut environ 20 minutes pour reconfigurer la carte réseau pour relancer le trafic.

La figure 48, montre que lundi le plafonnement105 de la bande passante à 9.2 Mbits/s a fonctionné correctement. Le plafonnement de la bande passante est employé pour prévenir par exemple des abus de téléchargements montant.

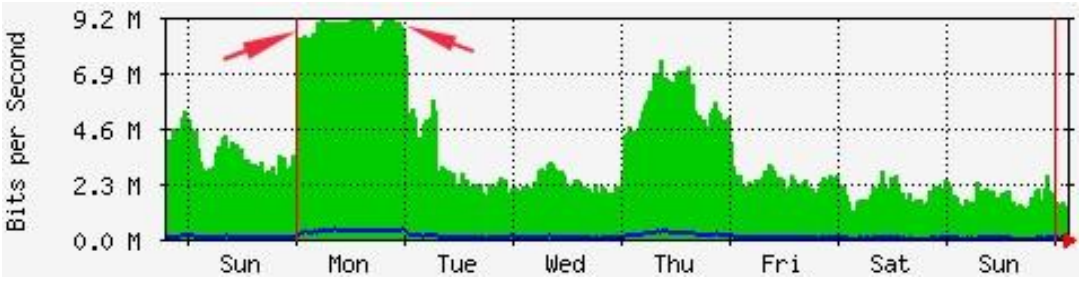


Figure 48 Plafonnement de la bande passante

Cette vue du trafic sur le réseau consiste à surveiller les sauvegardes quotidiennes, figure 49. Notons que ce graphique peut servir d'enregistrement en complément de la procédure de sauvegarde des données du SMSI.

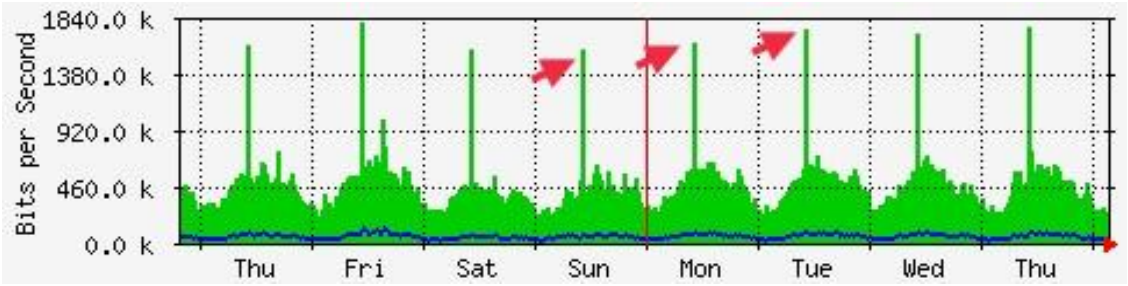


Figure 49 Trafic des sauvegardes quotidiennes

MRTG, est un outil de surveillance complet capable de restituer l'état du réseau en temps réel tout en gardant une trace des anciennes données.

MRTG et Nagios constituent une plate-forme logicielle complète, facilement modulable de supervision qui offre une grande variété de services répondant à tous les besoins de supervision.

Aujourd'hui, la communauté de Nagios rassemble plus de 250'000 utilisateurs et de nombreux développeurs. Parmi les dernières avancées, une refonte complète du cœur de Nagios en langage Python [41]. Ce langage apporte une nouvelle architecture plus souple et plus facile à maintenir que l'actuel démon.

[105] Le plafonnement est une technique qui consiste à limiter dans le temps le trafic sortant sur par exemple un serveur.

CONCLUSION

La sécurité de l'information est un élément majeur pour les entreprises qui souhaitent protéger leurs données. Sécuriser un réseau, un serveur, protéger des routeurs et bien configurer un pare-feu sont des techniques maîtrisées mais cette approche peut elle permettre d'apprécier les besoins réels et les priorités de l'entreprise en matière de sécurité ?

Pour un nombre croissant d'entreprises qui ont ces deux dernières décennies appliqué des méthodes de gestion des risques et des standards tels que le BS 7799-2, ces démarches parcellaires et souvent artisanales ne garantissaient pas une sécurité satisfaisante de l'information.

En adoptant les « meilleures pratiques » de sécurité qui leur étaient proposées par les standards et les méthodes telles que MEHARI, EBIOS ou OCTAVE, ces entreprises ont pu gérer la sécurité de l'information.

En 2005, l'ISO/CEI 27001, pilier d'une famille de normes encore en développement, a posé les bases du SMSI (Système de Management de la Sécurité de l'information). L'approche système, fondée sur le principe de l'amélioration continue à travers le cycle du PDCA est une étape essentielle dans l'évolution de la sécurité. En outre, ces nouvelles normes bénéficient d'une reconnaissance internationale contrairement aux standards et méthodes.

A partir de ce constat, Suss MicroOptics a considéré la mise en œuvre d'un SMSI. L'équipe du projet, constituée de trois personnes, a appliqué les deux premières phases du PDCA.

La première phase, « PLAN », nous a permis d'apprécier les risques liés aux actifs du périmètre du SMSI.

Définir le périmètre d'un SMSI exige de la concertation et beaucoup de réflexion puisque de cette première étape dépend l'étendue de l'analyse. La taille du site et le nombre de collaborateurs de SMO étant réduits, nous avons choisi d'inclure l'ensemble des actifs de SMO Neuchâtel au périmètre du SMSI en envisageant de

l'étendre à la maison mère dans le futur. Notons qu'une mauvaise évaluation du périmètre peut entrainer l'abandon de la mise en œuvre du SMSI.

Le SoM (statement of Maturity), état des lieux nous a permis de présenter à la Direction les écarts entre les mesures de sécurité de la norme et celles appliquées par SMO, donnant ainsi un aperçu plus précis du travail à entreprendre.

L'essentiel de l'analyse se situe au niveau de l'appréciation des risques, leur impact sur les actifs et le niveau de vulnérabilité de ces derniers. Notre RATP (Risk Assessment Treatment Plan) nous a permis d'établir méthodiquement les besoins et priorités en matière de sécurité.

Sur la base de cette analyse un DRP (Disaster Recovery Plan), de l'environnement TI de SMO a été produit. Il est intéressant de noter que l'idée d'implémenter un plan de continuité a été soutenue par un client de SMO, ce qui souligne la difficulté d'établir un périmètre pour le SMSI.

Enfin, un résumé des besoins du système d'information identifiant les contrôles choisis et déployés dans la phase « DO », vient achever la phase « PLAN ».

Parmi les mesures de sécurité à déployer, nous avons retenu la mise en œuvre d'une plate-forme logicielle de supervision pour répondre au besoin du « capacity management ».

Nous nous sommes appuyés sur les normes de l'ISO et les RFCs de l'IETF afin de cerner et comprendre les concepts et standards de la supervision.

Face à l'offre logicielle imposante de plate-formes et outils de supervision, nous avons fait une petite étude comparative des différents produits au terme de laquelle nous avons opté pour le moteur d'ordonnancement Nagios V3 sous licence GNU/GPL.

Bien que cette étude nous ait montré que « l'open source » n'était pas toujours la meilleure solution, l'utilisation de Nagios montre de multiples avantages pour SMO. Un des principaux est sa modularité qui permet de dimensionner l'application au plus près des besoins. Nagios peut être perçu comme le chef d'orchestre de la supervision autour duquel viennent s'intégrer des outils tels que les plugins « check_nt » « check_ping » ou les programmes comme « MRTG ».

La plate-forme logicielle de supervision nous a livré ses premiers enregistrements dès le mois de Mai 2010. Leur analyse permettra d'affiner les indicateurs de mesures et de dresser un premier bilan sur le travail entrepris et les ressouces allouées. Cette étape sera initiée par la revue de direction qui coïncide avec la phase « CHECK » du PDCA. La phase « ACT » de notre premier cycle correspondra aux améliorations qui seront apportées au système. On estime à une année la période de rodage nécessaire au bon fonctionnement d'un système de management.

BIBLIOGRAPHIE

[1] ISO. Discover ISO. Disponible sur : http://www.iso.org/iso/home.htm (consulté le 10/03/10).

[2] CEN. About us. Disponible sur : http://www.cen.eu/cen/pages/default.aspx (consulté le 10/03/10).

[3] Alan Calder, The case for ISO 27001. Ed. IT Governance Publishing 2006 p.80-83.

[4] Jean-Louis Le Moigne, La Théorie du système général: Théorie de la modélisation. Presses Universitaires de France, Paris, 1977.

[5] ISO 9001. Quality management systems : Requirements. International Organization for Standardization, Geneva, 2000.

[6] Alan Calder, Information Security Based on ISO 27001/ISO 27002 A Management Guide Editeur : van Haren Publishing; Édition : 2nd edition (31 juillet 2009) p.23-26

[7] ISO. Information security management systems. Disponible sur : http://www.iso.org, (consulté le 15/03/10).

[8] ISO/IEC 27002. Information technology - Security techniques - Code of practice for information security management. International Organization for Standardization, Geneva, 2005.

[9] ISO/IEC 27005. Information technology - Security techniques - Information security risk management. International Organization for Standardization, Geneva, 2008.

[10] ISO/IEC 27001. Information technology - Security techniques - Information security management systems - Requirements. International Organization for Standardization, Geneva, 2005.

[11] ENISA Inventory of Risk Management / Risk Assessment Methods and Tools. Disponible sur : http://www.enisa.europa.eu/ (consulté le 22/03/10).

[12] EBIOS, Méthode de gestion des risques, ANSSI, Version du 25 janvier 2010.

[13] CLUSIF, Présentation de MEHARI. Disponible sur : http://www.clusif.asso.fr/ (consulté le 23/03/10).

[14] C. Alberts, A. Dorofee, J. Stevens. Introduction to the OCTAVE® Approach. Networked Systems Survivability Program. Carnegie Mellon Software Engineering Institute Pittsburgh, 2003.

[15] ISO/IEC 27001. Information technology - Security techniques - Information security management systems - Requirements. International Organization for Standardization, Geneva, 2005 (clauses 4.2.1.c.2 et 5.1.f).

[16] Patrick Ow, Risk Management Guide, Managing Business Continuity Risk. Disponible sur : http://www.scribd.com, (consulté le 20/05/10).

[17] François Pignet, Réseaux Informatiques - Supervision et Administration. Ed. ENI 2007.

[18] Douglas R. Mauro, Kevin J. Schmidt. Essential Snmp. Ed. O'Reilly Media, Éd. 2nd Revised 2005.

[19] SNMP Research International Inc. SNMP RFCs. Disponible sur : http://www.snmp.org, (consulté le 10/02/10).

[20] Wikipédia. PING (logiciel). Disponible sur : http://fr.wikipedia.org, (consulté le 20/06/10).

[21] Wikipédia. Traceroute. Disponible sur : http://fr.wikipedia.org, (consulté le 20/06/10).

[22] Wikipédia. Netstat. Disponible sur : http://fr.wikipedia.org, (consulté le 20/06/10).

[23] Wikipédia. RRDTool. Disponible sur : http://fr.wikipedia.org, (consulté le 20/06/10).

[24] Cacti official forum. What is Cacti ? Disponible sur : http://www.cacti.net, (consulté le 22/02/10).

[25] Zabbix. DoKuWiki. Disponible sur : http://www.zabbix.com, (consulté le 22/02/10).

[26] OpenNMS. About OpenNMS. Disponible sur : http://www.opennms.org, (consulté le 21/02/10).

[27] Centreon. Centreon Wiki. Disponible sur : http://www.centreon.com, (consulté le 21/02/10).

[28] Taylor Dondich. Network Monitoring with Nagios. Ed. O'Reilly Media, 2006.

[29] Tobi Oetiker. What is MRTG ? Disponible sur : http://oss.oetiker.ch/mrtg, (consulté le 15/05/10).

[30] Carla Schroder. Linux Networking Cookbook. Ed. O'Reilly Media, 2007.

[31] Tobi Oetiker. Creates mrtg.cfg files (for mrtg-2.16.2). Disponible sur : http://oss.oetiker.ch/mrtg, (consulté le 15/05/10).

[32] Tobi Oetiker. Creates index files for mrtg web sites (mrtg-2.16.2). Disponible sur : http://oss.oetiker.ch/mrtg, (consulté le 15/05/10).

[33] Netfilter. What is iptables ? Disponible sur : http://www.netfilter.org, (consulté le 18/05/10).

[34] Nagios. Nagios Plugins. Disponible sur : http://nagiosplugins.org, (consulté le 19/05/10).

[35] Nagios. Developer-guidelines. Disponible sur : http://nagiosplugins.org, (consulté le 19/05/10).

[36] Ethan Galstad. NRPE documentation. Disponible sur : http://nagios.sourceforge.net, (consulté le 19/05/10).

[37] Nagios. NSCA with Nagios. Disponible sur : http://nagios.sourceforge.net, (consulté le 19/05/10).

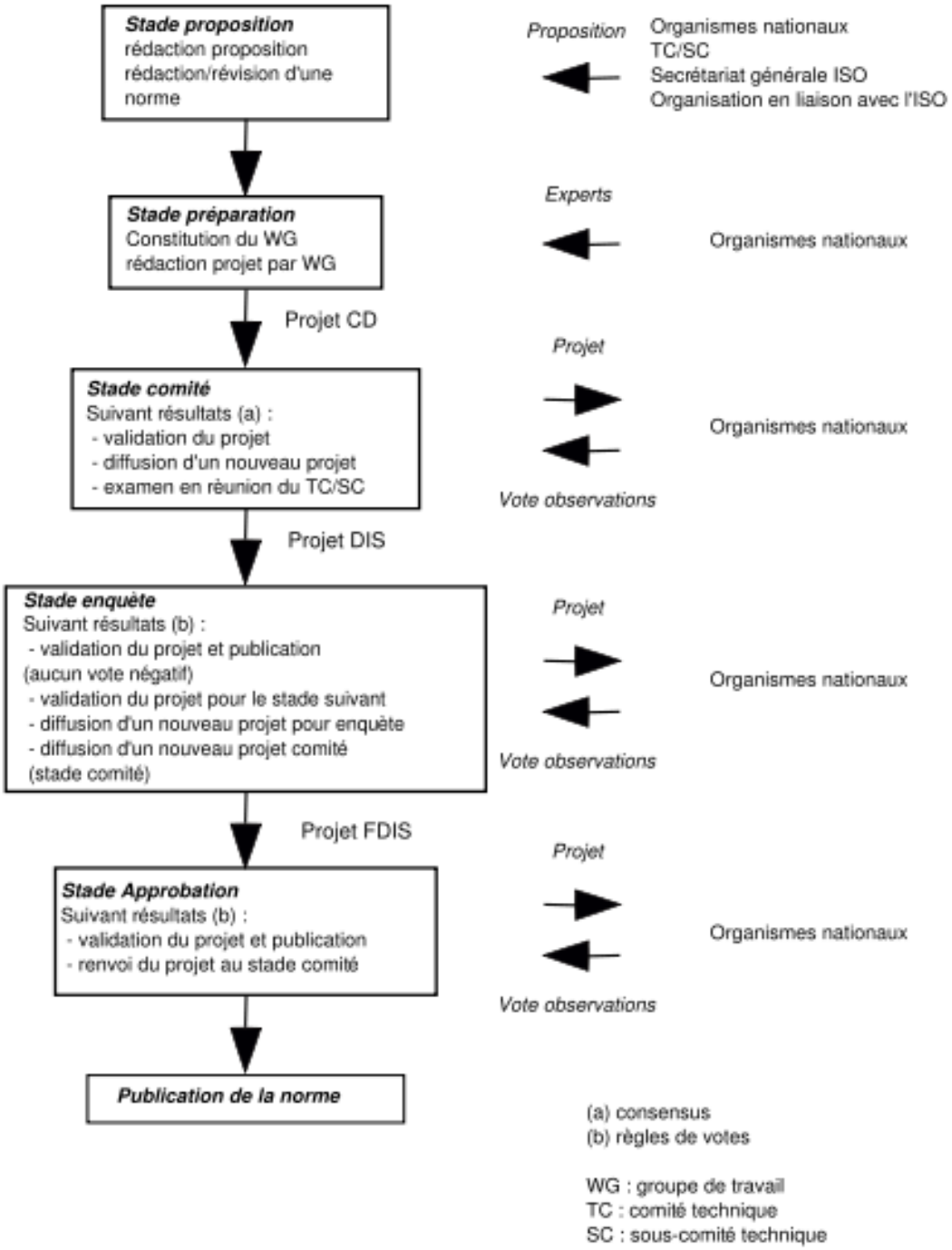
[38] NSClient++. About NSClient++. Disponible sur : http://nsclient.org/nscp/, (consulté le 19/05/10).

[40] Kioslea. How email works (MTA, MDA, MUA). Disponible sur : http://en.kioskea.net, (consulté le 09/06/10).

[41] Nagios. Nagios Community. Disponible sur : http://network.nagios.org/, (consulté le 15/06/10).

TABLE DES ANNEXES

Annexe 1. Processus d'élaboration des normes suivant l'ISO/CEI.



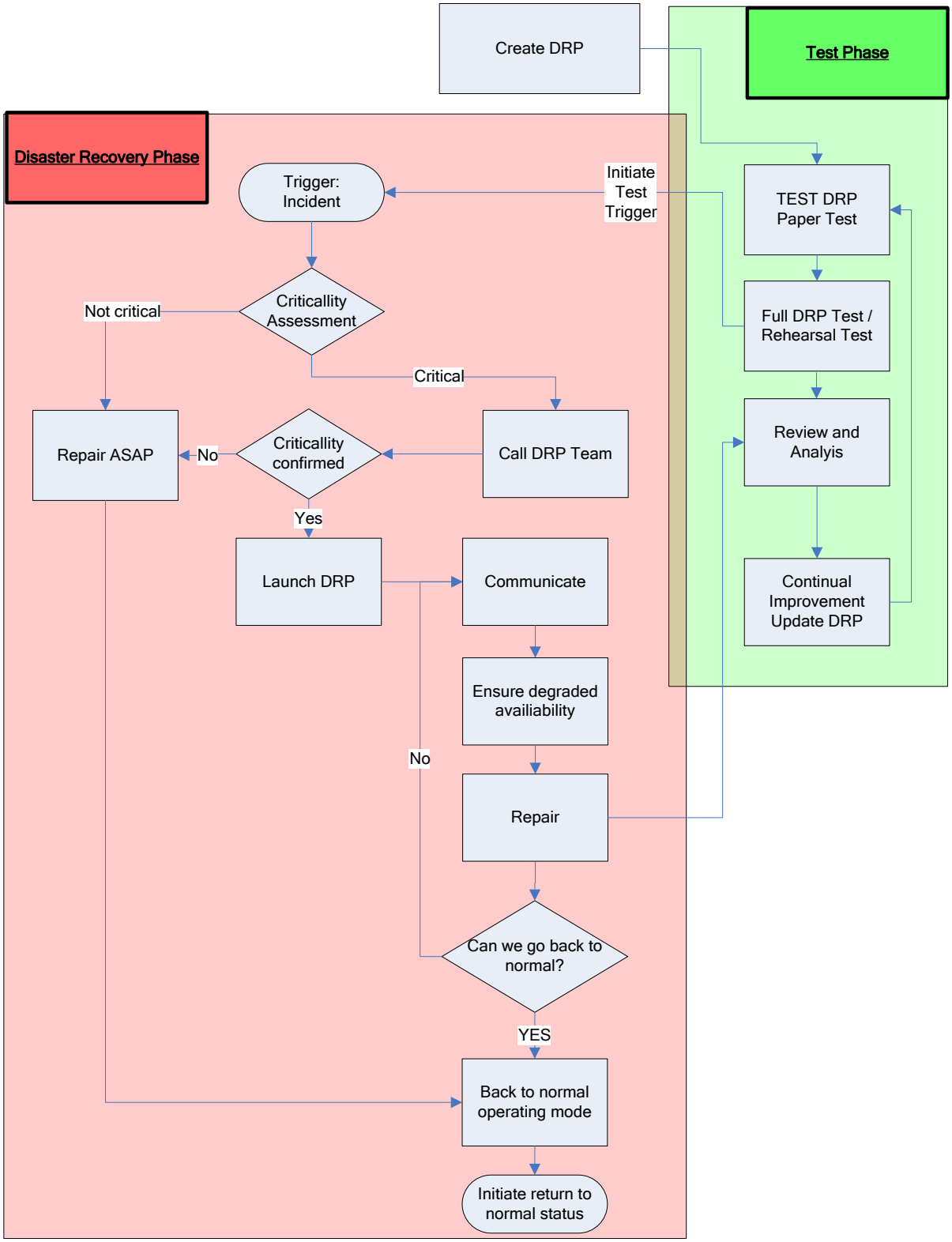
Annexe 2. DRP (Disaster Recovery Plan) Applications.

Application	***Impact / Risk***	***Mitigation***	***Category***	***DRP Status***
Production application	Stop of production		Critical	TBI
Microsoft Outlook	Unable to read emails / Loose of e-mails	.PST files stored on files servers / Reinstall and configure application	Significant	TBI
Microsoft Office (Winword, Excel, Powerpoint)	Unable to edit documents	Reinstall and configure application	Low	I
Microsoft Office Visio	Unable to edit flowcharts	Reinstall and configure application	Low	I
Adobe Acrobat Standard	Unable to create Acrobat files	Reinstall and configure application	Low	I
McAfee ePolicy Orchestrator	Computer are not protected against viruses and malwares	Reinstall and configure application	Significant	I
McAfee Agent	Computer are not protected against viruses and malwares	Reinstall and configure application	Significant	I

Annexe 3. DRP (Disaster Recovery Plan) IT Systems.

IT Systems	***Impact / Risk***	***Mitigation***	***Category***	***DRP Status***
File Server	Users can't retrieve their critical documents and users cannot share their data. Business will be affected.	Backup policy in place.	Critical	I
LAN infrastructure	Users can't connect to local servers and data.	Contact Uditis, the company which is responsible for LAN infrastructure.	Critical	TBI
WAN infrastructure	Users can't access to the Internet.	Access possible for laptop throught 3G. Contact the Internet provider.	Critical	TBI
DHCP server	No access to network.	Fix DHCP servers. Use IP fixe addresses.	Critical	TBI
Production PC	Stop of production.		Critical	TBI
Backup device	Recovery of key data will not be possible.		Critical	TBI
Telephony System		Use mobile phones for outgoing communication.	Significant	TBI
Physical Access & Security system	Unauthorized people might access to some resources. Security to area may be compromised.		Low	TBI
DNS Server	User can't easily access to computers in the network.	Fix DNS Server. Use IP addresses instead of names to access to computers.	Critical	TBI
Active Directory	Users can't logon.	Fix Active Directory. Logon locally on computers.	Significant	TBI
VPN	Users can't access to the company's resources from external locations.	Fix Remote Access Server. Users can access to their emails through the webmail.	Low	TBI

Annexe 4. DRP (Disaster Recovery Plan) Processus.



Annexe 5. Tableau comparatif des plateformes logicielles de supervision**.**

Comparison of network monitoring software

Name	IP SLA Reports	Logical Grouping	Trending	Trend Prediction	Auto Discovery	Agent	SNMP	Syslog	Plugins	Triggers / Alerts	WebApp	Distributed Monitoring	Inventory	Data Storage Method	License	Maps	Access Control	IPv6
AccelOps	Yes	Yes	Yes	Yes	Yes	Supported	Yes	Yes	Yes	Yes	Full Control	Yes	Yes	PostgreSQL, in memory, proprietary event DB	Commercial	Yes	Yes	Unknown
Argus	Yes	Yes	No	No	No	Yes	Yes	Yes	Yes	Yes	Viewing, Acknowledging	Yes	Unknown	Berkeley DB	Artistic License	No	Granular	Yes
Cacti	Yes	Yes	Yes	Yes	Via plugin	No	Yes	Yes	Yes	Yes	Full Control	Yes	Yes	RRDtool, MySQL	GPL	Plugin	Yes	Yes
CiscoWorks LMS	Yes	Yes	Unknown	No	Yes	No	Yes	Yes	Unknown	Yes	Viewing, Acknowledging, Reporting	Yes	Yes	Unknown	Commercial	Yes	Yes	Yes
collectd	No	No	No	No	Push model; multicast possible	Supported	Yes	Yes	Yes	Yes	Viewing	Yes	No	RRDtool, CSV, in memory, plugins	GPLv2	No	Apache ACL	Yes
Dhyan Network management System	Yes	Yes	Yes	Unknown	Yes	Supported	Yes	Yes	Yes	Yes	Fullcontrol	Yes	Yes	MySQL, Oracle, Derby	Commercial	Yes	Yes	Yes
dopplerVUE	Yes	Yes	Yes	No	Yes	Yes	Yes	Yes	Yes	Yes	Full Control	Yes	Yes	MS SQL	Commercial	Yes	Granular	Yes
FreeNATS	Yes	Yes	No	No	Yes	Yes	No	Via plugin	Yes	In PHP Code	Full Control	No	No	MySQL	GPL	No	Granular	Unknown
Ganglia	No	Yes	Yes	No	Via gmond check in	Yes	Via plugin	No	Yes	No	Viewing	Yes	Unknown	RRDtool, in memory	BSD	Yes	No	Unknown
Intellipool Network Monitor	Yes	Yes	No	No	Yes	No	Yes	Yes	Yes	No	Viewing, Acknowledging, Reporting	Yes	Yes	FirebirdSQL	Commercial	Yes	Granular	Unknown
AdRem NetCrunch	No	Yes	Yes	No	Yes	No	Yes	Yes	Yes	Yes	Viewing, Acknowledging	No	No	SQL	Commercial	Yes	Yes	Unknown
IPHost Network Monitor	Yes	Yes	Yes	No	Yes	No	Yes	No	Yes	Yes	Viewing, Acknowledging, Reporting	No	No	FirebirdSQL	Commercial	No	No	Unknown
Munin	No	No	Yes	Unknown	No	Yes	Yes	No	Yes	Partial	Viewing	Unknown	Unknown	RRDtool	GPL	Unknown	Unknown	Yes
Nagios	Via plugin	Yes	Yes	No	Via plugin	Yes	Via plugin	Via plugin	Yes	Yes	Full Control	Yes	Via plugin	Flat file, SQL	GPL	Yes	Yes	Yes
NetMRI	No	Yes	Yes	No	Yes	No	Yes	Yes	Yes	Yes	No	Yes	Yes	MySQL	Commercial	Yes	Yes	Unknown
NetQoS Performance Center	Yes	Yes	Yes	Yes	Yes	No	Yes	Yes	Yes	Yes	Viewing, Acknowledging, Reporting	Yes	Yes	Yes	Commercial	Yes	Yes	Unknown
Nimsoft Monitoring Solution	Yes	Yes	Yes	Yes	Yes	Supported	Yes	Yes	Yes	Yes	Viewing, Acknowledging, Reporting	Yes	Yes	SQL	Commercial	Yes	Yes	Unknown

Annexe 5 (suite). Tableau comparatif des plateformes logicielles de supervision**.**

Name	IP SLA Reports	Logical Grouping	Trending	Trend Prediction	Auto Discovery	Agent	SNMP	Syslog	Plugins	Triggers / Alerts	WebApp	Distributed Monitoring	Inventory	Data Storage Method	License	Maps	Access Control	IPv6
OpenNMS	Yes	Yes	Yes	Unknown	Yes	Supported	Yes	Yes	Yes	Yes	Full Control	Yes	Limited	JRobin, PostgreSQL [1]	GPL	Yes	Yes	Limited
OPNET ACE Live	Yes	Yes	Yes	Yes	Yes	No	Yes	No	Yes	Yes	Viewing, Acknowledging, Reporting	Yes	No	Yes	Commercial	Yes	Yes	Unknown
Opsview	Yes	Yes	Yes	No	Yes	Yes	Yes	Yes	Yes	Yes	Full Control	Yes	No	SQL	GPL	Yes	Granular	Yes
PacketTrap	No	Yes	Yes	Unknown	Yes	Yes	Yes	Yes	Yes	Yes	Viewing, Reporting	Yes	Unknown	SQL	Commercial	Unknown	Unknown	Unknown
Pandora FMS	Yes	Yes	Yes	Yes	Yes	Supported	Yes	Yes	Yes	Yes	Full Control	Yes	Yes	MySQL	GPLv2; (Enterprise edition available)	Yes	Granular	Unknown
Performance Co-Pilot	No	Yes	Yes	No	No	Yes	No	No	Yes	Yes	No	Yes	No	Flat file	GPL, LGPL	No	No	Unknown
PRTG Network Monitor	Yes	Yes	Yes	Yes	Yes	Supported	Yes	Yes	Yes	Yes	Full Control	Yes	Yes	SQL	Freeware and Commercial	Yes	Granular	Unknown
Scrutinizer	Yes	Yes	Yes	No	No	No	Yes	Yes	Yes	Yes	Viewing, Acknowledging, Reporting	Yes	Yes	MySQL	Commercial	Yes	Yes	Yes
ServersCheck	Yes	Yes	Yes	No	Yes	Supported	Yes	Yes	Yes	Yes	Full Control	Yes	No	Flat file, ODBC	Commercial	Yes	Yes	Unknown
SevOne	Yes	Yes	Yes	Yes	Yes	No	Yes	No	Yes	Yes	Full Control	Yes	Unknown	MySQL	Commercial	Yes	Yes	Unknown
Orion	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Full Control	Yes	Yes	SQL	Commercial	Yes	Yes	Yes
Shinken	Via plugin	Yes	Yes	No	Via plugin	Yes	Via plugin	Via plugin	Yes	Yes	Viewing, Acknowledging, Reporting	Yes	Via plugin	Flat file, MySQL, Oracle, CouchDB, Sqlite	AGPL	Yes	No	Unknown
Spiceworks	Yes	Yes	No	No	Yes	Supported	Yes	No	Yes	Yes	Viewing, Acknowledging, Reporting	Yes	Yes	Sqlite	Commercial (Free)	Yes	Admin & Report user, More via plugins	Unknown
TclMon	Yes	Yes	Yes	No	Yes	Supported	Yes	Yes	Yes	Yes	Viewing	No	No	RRDTool, in memory, plugins	BSD	Yes	Yes	No
Zabbix	Yes	Yes	Yes	Yes	Yes	Supported	Yes	Yes	Yes	Yes	Full Control	Yes	Yes	Oracle, MySQL, PostgreSQL, SQLite	GPL	Yes	Yes	Yes
Zenoss	Yes	Yes	Yes	Yes	Yes	No	Yes	Yes	Yes	Yes	Full Control	Yes	Yes	ZODB, MySQL, RRDtool	GPL Zenoss Core; paid Pro and Enterprise editions [2]	Yes	Yes	Unknown
Zyrion Traverse	Yes	Yes	Yes	Yes	Yes	Supported	Yes	Yes	Yes	Yes	Full Control	Yes	Yes	SQL	Commercial	Yes	Multi-tier	Unknown

Annexe 6. Extrait du code source de Nagios.

```
/*****************************************************************

 * NAGIOS.C - Core Program Code For Nagios
 *
 * Program: Nagios Core
 * Version: 3.2.1
 * License: GPL
 * Copyright (c) 2009-2010 Nagios Core Development Team and Community Contributors
 * Copyright (c) 1999-2009 Ethan Galstad
 *
 * First Written:   01-28-1999 (start of development)
 * Last Modified:   03-09-2010
 *
 * Description:
 *
 * Nagios is a network monitoring tool that will check hosts and services
 * that you specify.  It has the ability to notify contacts via email, pager,
 * or other user-defined methods when a service or host goes down and
 * recovers.  Service and host monitoring is done through the use of external
 * plugins which can be developed independently of Nagios.
 *
 * License:
 *
 * This program is free software; you can redistribute it and/or modify
 * it under the terms of the GNU General Public License version 2 as
 * published by the Free Software Foundation.
 *
 * This program is distributed in the hope that it will be useful,
 * but WITHOUT ANY WARRANTY; without even the implied warranty of
 * MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
 * GNU General Public License for more details.
 *
 * You should have received a copy of the GNU General Public License
 * along with this program; if not, write to the Free Software
 * Foundation, Inc., 675 Mass Ave, Cambridge, MA 02139, USA.
 *
 *****************************************************************************/

#include "../include/config.h"
#include "../include/common.h"
#include "../include/objects.h"
#include "../include/comments.h"
#include "../include/downtime.h"
#include "../include/statusdata.h"
#include "../include/macros.h"
#include "../include/nagios.h"
#include "../include/sretention.h"
#include "../include/perfdata.h"
#include "../include/broker.h"
#include "../include/nebmods.h"
#include "../include/nebmodules.h"

/*#define DEBUG_MEMORY 1*/
#ifdef DEBUG_MEMORY
#include <mcheck.h>
#endif


char           *config_file=NULL;
char           *log_file=NULL;
char            *command_file=NULL;
char            *temp_file=NULL;
char            *temp_path=NULL;
char            *check_result_path=NULL;
char            *lock_file=NULL;
char            *log_archive_path=NULL;
char            *p1_file=NULL;    /**** EMBEDDED PERL ****/
char            *auth_file=NULL;  /**** EMBEDDED PERL INTERPRETER AUTH FILE ****/
char            *nagios_user=NULL;
char            *nagios_group=NULL;
```

```
extern char     *macro_x[MACRO_X_COUNT];

char            *global_host_event_handler=NULL;
char            *global_service_event_handler=NULL;
command         *global_host_event_handler_ptr=NULL;
command         *global_service_event_handler_ptr=NULL;

char            *ocsp_command=NULL;
char            *ochp_command=NULL;
command         *ocsp_command_ptr=NULL;
command         *ochp_command_ptr=NULL;

char            *illegal_object_chars=NULL;
char            *illegal_output_chars=NULL;

int             use_regexp_matches=FALSE;
int             use_true_regexp_matching=FALSE;

int            use_syslog=DEFAULT_USE_SYSLOG;
int             log_notifications=DEFAULT_NOTIFICATION_LOGGING;
int             log_service_retries=DEFAULT_LOG_SERVICE_RETRIES;
int             log_host_retries=DEFAULT_LOG_HOST_RETRIES;
int             log_event_handlers=DEFAULT_LOG_EVENT_HANDLERS;
int             log_initial_states=DEFAULT_LOG_INITIAL_STATES;
int             log_external_commands=DEFAULT_LOG_EXTERNAL_COMMANDS;
int             log_passive_checks=DEFAULT_LOG_PASSIVE_CHECKS;
```

Annexe 7. Extrait du plugin Nagios Check_nt.

```
/*****************************************************************************
*
* Nagios check_nt plugin
*
* License: GPL
* Copyright (c) 2000-2002 Yves Rubin (rubiyz@yahoo.com)
* Copyright (c) 2003-2007 Nagios Plugins Development Team
*
* Description:
*
* This file contains the check_nt plugin
*
* This plugin collects data from the NSClient service running on a
* Windows NT/2000/XP/2003 server.
* This plugin requires NSClient software to run on NT
* (http://nsclient.ready2run.nl/)
*
*
* This program is free software: you can redistribute it and/or modify
* it under the terms of the GNU General Public License as published by
* the Free Software Foundation, either version 3 of the License, or
* (at your option) any later version.
*
* This program is distributed in the hope that it will be useful,
* but WITHOUT ANY WARRANTY; without even the implied warranty of
* MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
* GNU General Public License for more details.
*
* You should have received a copy of the GNU General Public License
* along with this program.  If not, see <http://www.gnu.org/licenses/>.
*
*
*****************************************************************************/

const char *progname = "check_nt";
const char *copyright = "2000-2007";
const char *email = "nagiosplug-devel@lists.sourceforge.net";

#include "common.h"
#include "netutils.h"
#include "utils.h"

enum checkvars {
       CHECK_NONE,
       CHECK_CLIENTVERSION,
       CHECK_CPULOAD,
       CHECK_UPTIME,
       CHECK_USEDDISKSPACE,
       CHECK_SERVICESTATE,
       CHECK_PROCSTATE,
       CHECK_MEMUSE,
       CHECK_COUNTER,
       CHECK_FILEAGE,
       CHECK_INSTANCES
};

enum {
       MAX_VALUE_LIST = 30,
       PORT = 1248
};

char *server_address=NULL;
char *volume_name=NULL;
int server_port=PORT;
char *value_list=NULL;
char *req_password=NULL;
unsigned long lvalue_list[MAX_VALUE_LIST];
unsigned long warning_value=0L;
unsigned long critical_value=0L;
int check_warning_value=FALSE;
int check_critical_value=FALSE;
enum checkvars vars_to_check = CHECK_NONE;
int show_all=FALSE;
```

```
char recv_buffer[MAX_INPUT_BUFFER];

void fetch_data (const char* address, int port, const char* sendb);
int process_arguments(int, char **);
void preparelist(char *string);
int strtoularray(unsigned long *array, char *string, const char *delim);
void print_help(void);
void print_usage(void);

int main(int argc, char **argv){

/* should be   int result = STATE_UNKNOWN; */

        int return_code = STATE_UNKNOWN;
        char *send_buffer=NULL;
        char *output_message=NULL;
        char *perfdata=NULL;
        char *temp_string=NULL;
        char *temp_string_perf=NULL;
        char *description=NULL,*counter_unit = NULL;
        char *minval = NULL, *maxval = NULL, *errcvt = NULL;
        char *fds=NULL, *tds=NULL;

        double total_disk_space=0;
        double free_disk_space=0;
        double percent_used_space=0;
        double warning_used_space=0;
        double critical_used_space=0;
        double mem_commitLimit=0;
        double mem_commitByte=0;
        double fminval = 0, fmaxval = 0;
        unsigned long utilization;
        unsigned long uptime;
        unsigned long age_in_minutes;
        double counter_value = 0.0;
        int offset=0;
        int updays=0;
        int uphours=0;
        int upminutes=0;

        int isPercent = FALSE;
        int allRight = FALSE;

        setlocale (LC_ALL, "");
        bindtextdomain (PACKAGE, LOCALEDIR);
        textdomain (PACKAGE);

        /* Parse extra opts if any */
        argv=np_extra_opts (&argc, argv, progname);

        if(process_arguments(argc,argv) == ERROR)
                usage4 (_("Could not parse arguments"));

        /* initialize alarm signal handling */
        signal(SIGALRM,socket_timeout_alarm_handler);

        /* set socket timeout */
        alarm(socket_timeout);

        switch (vars_to_check) {

        case CHECK_CLIENTVERSION:

                asprintf(&send_buffer, "%s&1", req_password);
                fetch_data (server_address, server_port, send_buffer);
                if (value_list != NULL && strcmp(recv_buffer, value_list) != 0) {
                        asprintf (&output_message, _("Wrong client version - running: %s,
required: %s"), recv_buffer, value_list);
                        return_code = STATE_WARNING;
                } else {
                        asprintf (&output_message, "%s", recv_buffer);
                        return_code = STATE_OK;
                }
```

Annexe 8. Extrait du plugin Nagios check_ping.

```
/*****************************************************************************
*
* Nagios check_ping plugin
*
* License: GPL
* Copyright (c) 2000-2007 Nagios Plugins Development Team
*
* Description:
*
* This file contains the check_ping plugin
*
* Use the ping program to check connection statistics for a remote host.
*
*
* This program is free software: you can redistribute it and/or modify
* it under the terms of the GNU General Public License as published by
* the Free Software Foundation, either version 3 of the License, or
* (at your option) any later version.
*
* This program is distributed in the hope that it will be useful,
* but WITHOUT ANY WARRANTY; without even the implied warranty of
* MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
* GNU General Public License for more details.
*
* You should have received a copy of the GNU General Public License
* along with this program.  If not, see <http://www.gnu.org/licenses/>.
*
*
*****************************************************************************/

const char *progname = "check_ping";
const char *copyright = "2000-2007";
const char *email = "nagiosplug-devel@lists.sourceforge.net";

#include "common.h"
#include "netutils.h"
#include "popen.h"
#include "utils.h"

#define WARN_DUPLICATES "DUPLICATES FOUND! "
#define UNKNOWN_TRIP_TIME -1.0       /* -1 seconds */

enum {
       UNKNOWN_PACKET_LOSS = 200,    /* 200% */
       DEFAULT_MAX_PACKETS = 5       /* default no. of ICMP ECHO packets */
};

int process_arguments (int, char **);
int get_threshold (char *, float *, int *);
int validate_arguments (void);
int run_ping (const char *cmd, const char *addr);
int error_scan (char buf[MAX_INPUT_BUFFER], const char *addr);
void print_usage (void);
void print_help (void);

int display_html = FALSE;
int wpl = UNKNOWN_PACKET_LOSS;
int cpl = UNKNOWN_PACKET_LOSS;
float wrta = UNKNOWN_TRIP_TIME;
float crta = UNKNOWN_TRIP_TIME;
char **addresses = NULL;
int n_addresses = 0;
int max_addr = 1;
int max_packets = -1;
int verbose = 0;

float rta = UNKNOWN_TRIP_TIME;
int pl = UNKNOWN_PACKET_LOSS;

char *warn_text;
int
main (int argc, char **argv)
```

```
{
        char *cmd = NULL;
        char *rawcmd = NULL;
        int result = STATE_UNKNOWN;
        int this_result = STATE_UNKNOWN;
        int i;

        setlocale (LC_ALL, "");
        setlocale (LC_NUMERIC, "C");
        bindtextdomain (PACKAGE, LOCALEDIR);
        textdomain (PACKAGE);

        addresses = malloc (sizeof(char*) * max_addr);
        addresses[0] = NULL;

        /* Parse extra opts if any */
        argv=np_extra_opts (&argc, argv, progname);

        if (process_arguments (argc, argv) == ERROR)
                usage4 (_("Could not parse arguments"));

        /* Set signal handling and alarm */
        if (signal (SIGALRM, popen_timeout_alarm_handler) == SIG_ERR) {
                usage4 (_("Cannot catch SIGALRM"));
        }
```

Annexe 9. Extrait du programme MRTG de Tobi Oetiker.

```
! /usr/bin/perl -w
# -*- mode: cperl -*-

##################################################################
# MRTG 2.16.4  Multi Router Traffic Grapher
##################################################################
# Created by Tobias Oetiker <tobi@oetiker.ch>
#            and Dave Rand <dlr@bungi.com>
#
# For individual Contributers check the CHANGES file
#
##################################################################
#
# Distributed under the GNU General Public License
#
##################################################################
my @STARTARGS=($0,@ARGV);

@main::DEBUG=qw();
# DEBUG TARGETS
# cfg  - watch the config file reading
# dir  - directory mangeling
# base - basic program flow
# tarp - target parser
# snpo - snmp polling
# snpo2 - more snmp debug
# coca - confcache operations
# repo - track confcache repopulation
# fork - forking view
# time - some timing info
# log  - logging of data via rateup or rrdtool
# eval - trace eval experssions
# prof - add timeing info some interesting bits of code

$main::GRAPHFMT="png";
# There older perls tend to behave peculiar with
# large integers ...
require 5.005;

use strict;
# addon jpt
BEGIN {
    # Automatic OS detection ... do NOT touch
    if ( $^O =~ /^(ms)?(dos|win(32|nt)?)/i ) {
       $main::OS = 'NT';
       $main::SL = '\\';
       $main::PS = ';';
    } elsif ( $^O =~ /^NetWare$/i ) {
       $main::OS = 'NW';
       $main::SL = '/';
       $main::PS = ';';
    } elsif ( $^O =~ /^VMS$/i ) {
       $main::OS = 'VMS';
       $main::SL = '.';
       $main::PS = ':';
    } elsif ( $^O =~ /^os2$/i ) {
       $main::OS = 'OS2';
       $main::SL = '/';
       $main::PS = ';';
    } else {
       $main::OS = 'UNIX';
       $main::SL = '/';
       $main::PS = ':';
    }
    if ( $ENV{LANG} and $ENV{LANG} =~ /UTF.*8/i ){
        my $args = join " ", map { /\s/ ? "\"$_\"" : $_ } @ARGV;
        $args ||= "";
        print <<ERR;
------------------------------------------------------------------------
ERROR: Mrtg will most likely not work properly when the environment
       variable LANG is set to UTF-8. Please run mrtg in an environment
       where this is not the case. Try the following command to start:
```

```
       env LANG=C ${0} $args
------------------------------------------------------------------------
ERR
        exit 0;
    }
}


use FindBin;
use lib "${FindBin::Bin}";
use lib "${FindBin::Bin}${main::SL}..${main::SL}lib${main::SL}mrtg2";
use Getopt::Long;
use Math::BigFloat;

# search for binaries in the bin and bin/../lib  directory
use MRTG_lib "2.100016";

my $NOW = timestamp;

# $SNMP_Session::suppress_warnings = 2;
use locales_mrtg "0.07";

# Import IPv6 libraries if available, as (unfortunately) they are
# necessary for dead host detection (for IPv6 name lookups).
BEGIN {
    if (eval {local $SIG{__DIE__};require Socket6;})  {
       import Socket;
       import Socket6
    }
}

# Do not Flash Console Windows for the forked rateup process
BEGIN {
    if ($^O eq 'MSWin32'){
       eval {local $SIG{__DIE__};require Win32; Win32::SetChildShowWindow(0)};
       warn "WARNING: $@\n" if $@;
    }
}

$main::STARTTIME = time;


if ($MRTG_lib::OS eq 'OS2') {
# in daemon mode we will pause 3 seconds to be sure that parent died
  require OS2::Process;
  if (OS2::Process::my_type() eq 'DETACH') {sleep(3);}
}

if ($MRTG_lib::OS eq 'UNIX') {
   $SIG{INT} = $SIG{TERM} =
            sub {   unlink ${main::Cleanfile}
                        if defined $main::Cleanfile;
                    unlink ${main::Cleanfile2}
                        if defined $main::Cleanfile2;
                    unlink ${main::Cleanfile3}
                        if defined $main::Cleanfile3;
                    warn "$NOW: ERROR: Bailout after SIG $_[0]\n";
                    exit 1;
                };
  $SIG{HUP} = sub {
                    unlink ${main::Cleanfile}
                        if defined $main::Cleanfile;
                    unlink ${main::Cleanfile2}
                        if defined $main::Cleanfile2;
                    unlink ${main::Cleanfile3}
                        if defined $main::Cleanfile3;
                    die "$NOW: ERROR: Bailout after SIG $_[0]\n";
                };
}


END {
```

Annexe 10. Extrait du programme de MRTG de Dave Rand.

```
/*
 MRTG 2.16.4  -- Rateup
 *********************

 Rateup is a fast add-on to the great MRTG Traffic monitor.  It makes
 the database file updates much faster, and creates the graphic image
 files, ready for processing by PPMTOGIF.  It also reduces memory
 requirements by a factor of 10, and increases the speed of updates
 by a factor of at least 10.  This makes it feasible to run mrtg
 every 5 minutes.

 rateup attempts to compensate for missed updates by repeating the last
 sample, and also tries to catch bad update times.  The .log file stores
 real history every five minutes for 31 hours, then 'compresses' the
 history into 30 minute samples for a week, then 2-hour samples for
 31 days, then daily samples for two years.  This ensures that the
 log files don't grow in size.

 The log files are a slightly different format, but convert.perl
 will fix that for you.

 Enjoy!
 Dave Rand
 dlr@bungi.com

 04/26/99 - There was some compilation bug under Watcom 10.6
            which was fixed when recompiled with VC++ 6.0
                      Alexandre Steinberg
                      steinberg@base.com.br

*/

#include <stdlib.h>
#include <string.h>
/* VC++ does not have unistd.h */
#ifndef WIN32
#ifndef NETWARE
#include "../config.h"
#endif
#include <unistd.h>
#endif
#include <limits.h>
#include <stdio.h>
#include <time.h>
#include <math.h>
#include <ctype.h>
#ifndef GFORM_GD
#define GFORM_GD gdImagePng
#endif

/* BSD* does not have/need malloc.h */
#if !defined(bsdi) && !defined(__FreeBSD__) && !defined(__OpenBSD__) && !defined(__APPLE__)
#include <malloc.h>
#endif

/* MSVCRT.DLL does not know %ll in printf */
#ifdef __MINGW32_VERSION
#define LLD "%I64d"
#define LLD_FORMAT "I64d"
#endif

#ifdef __EMX__                  /* OS/2 */
#define strtoll _strtoll
#define LLD "%Ld"               /* EMX lib use %Ld for long long */
#define LLD_FORMAT "Ld"
#endif

#ifndef LLD
#define  LLD "%lld"
#define  LLD_FORMAT "lld"
#endif
```

```
/* WATCOM C/C++ 10.6 under Win95/NT */
/* VC++ 6.0 under Win95/NT */
#if defined(__WATCOMC__) || defined(WIN32)
#include <string.h>
#include <sys/types.h>
#include <direct.h>
#include <io.h>
#endif

#include <gd.h>
#include <gdfonts.h>

char *VERSION = "2.16.4";
char *program, *router, *routerpath;
int histvalid;

/* Options */
short options = 0;
#define OPTION_WITHZEROES     0x0001 /* withzeros */
#define OPTION_UNKNASZERO     0x0002 /* unknaszero */
#define OPTION_TRANSPARENT    0x0004 /* transparent */
#define OPTION_DORELPERCENT   0x0008 /* dorelpercent */
#define OPTION_NOBORDER              0x0010 /* noborder */
#define OPTION_NOARROW        0x0020 /* noarrow */
#define OPTION_NO_I           0x0040 /* ignore 'I' (first) variable */
#define OPTION_NO_O           0x0080 /* ignore 'O' (second) variable */
#define OPTION_PNGDATE        0x0100 /* show date & time in graph */
#define OPTION_PRINTROUTER    0x0200 /* show title in graph */
#define OPTION_LOGGRAPH              0x0400 /* Use a logarithmic Y axis */
#define OPTION_MEANOVER              0x0800 /* max Y = mean-above-the-mean */
#define OPTION_EXPGRAPH              0x1000 /* exponential scale (opposite of logscale) */

time_t NOW;

/* jpt, april 2006 : added 3 lines for date & time logging */
struct tm * stLocal;
time_t timestamp;
char bufftime[32];

char *short_si_def[] = { "", "k", "M", "G", "T" };
int kMGnumber = 4;
char **short_si = short_si_def;
char *longup = NULL;
char *shortup = NULL;
char *pngtitle = NULL;
char *pngdate = NULL;
char *rtimezone = NULL;
char weekformat = 'V';        /* strftime() fmt char for week #  */

#define DAY_COUNT (600)               /* 400 samples is 33.33 hours */
#define DAY_SAMPLE (5*60)     /* Sample every 5 minutes */
#define WEEK_COUNT (600)      /* 400 samples is 8.33 days */
#define WEEK_SAMPLE (30*60)   /* Sample every 30 minutes */
#define MONTH_COUNT (600)     /* 400 samples is 33.33 days */
#define MONTH_SAMPLE (2*60*60)        /* Sample every 2 hours */
#define YEAR_COUNT  (2 * 366) /* 1 sample / day, 366 days, 2 years */
#define YEAR_SAMPLE (24*60*60)        /* Sample every 24 hours */

/* One 'rounding error' per sample period, so add 4 to total and for
   good mesure we take 10 :-) */
#define MAX_HISTORY (DAY_COUNT+WEEK_COUNT+MONTH_COUNT+YEAR_COUNT+10)

/* These are the colors used for the variouse parts of the graph */
/* the format is Red,Green,Blue */

#define c_blank 245,245,245
#define c_light 194,194,194
#define c_dark 100,100,100
#define c_major 255,0,0
#define c_in    0,235,12
#define c_out   0,94,255
#define c_grid  0,0,0
#define c_inm   0,166,33
#define c_outm  255,0,255
#define c_outp  239,159,79
```

```
int col_in[3];
int col_out[3];
int col_inm[3];
int col_outm[3];
int col_outp[3];

long long kilo = (long long) 1000;
char *kMG = (char *) NULL;


#define MAXL   200            /* Maximum length of last in & out fields */


struct HISTORY
{
  time_t time;
  double in, out, percent, inmax, outmax;
}
 *history;
int Mh;

struct LAST
{
  time_t time;
  char in[MAXL], out[MAXL];
}
last;

#ifndef max
#define max(a,b) ((a) > (b) ? (a) : (b))
#endif
#ifndef min
#define min(a,b) ((a) < (b) ? (a) : (b))
#endif

/*

notes about the NEXT macro ....

* position n to the entry in the history array so that NOW is between
  history[n] and history[n+1]

* calculate the interval according to steptime and position of
  now within the history array.

for debuging

    fprintf (stderr,"%s, NOW: %8lu  ST: %4lu  N: %4u HTN: %8lu HTN+1: %8lu IV: %6.0f\n", \
            bufftime,now,steptime,n,history[n].time, history[n+1].time, interval);\

*/

#define NEXT(steptime) \
  inmax = outmax = avc = 0; \
  inr = outr = 0.0;\
  nextnow = now - steptime;\
  if (now == history[n].time && \
      n<histvalid && \
      nextnow == history[n+1].time) { \
    inr = (double) history[n].in;\
    outr = (double) history[n].out;\
    inmax = history[n].inmax;\
    outmax = history[n].outmax;\
    n++;\
  } else {\
    if (now > history[n].time) {\
      fprintf(stderr,"%s, ERROR: Rateup is trying to read ahead of the available data\n"
,bufftime);\
    } else {\
      while (now <= history[n+1].time && n < histvalid){n++;}\
      do {\
       if (now >= history[n].time) {\
         if (nextnow <= history[n+1].time) {\
           interval = history[n].time - history[n+1].time;\
         } else {\
           interval = history[n].time - nextnow;\
         }\
```

```
        } else {\
          if (nextnow > history[n+1].time) {\
            interval = steptime;\
          } else {\
            interval = now - history[n+1].time;\
          }\
        }\
        inr  += (double) history[n].in * interval;\
        outr += (double) history[n].out * interval;\
        avc += interval;\
        inmax =  (long long) max(inmax, history[n].inmax);\
        outmax = (long long) max(outmax,history[n].outmax);\
        if (nextnow <= history[n+1].time) n++; else break;\
      } while (n < histvalid && nextnow < history[n].time);\
\
      if (avc != steptime) {\
       fprintf(stderr,"%s, ERROR: StepTime does not match Avc %8" LLD_FORMAT ". Please Report
this.\n", bufftime, avc);\
      }\
\
      inr /= avc; outr /= avc;\
    }\
  }
```

LISTE DES FIGURES

LISTE DES TABLEAUX

Mise en oeuvre de la SSI (Sécurité du Système d'information) de SUSS MicroOptics par l'approche processus ISO/CEI 27001

Mémoire d'Ingénieur CNAM, Lyon 2010.

RESUME

Ce mémoire traite de la mise en œuvre d'un SMSI (système de management de la sécurité de l'information) dans une PME hautement spécialisée dans le domaine de la Micro Optique. Après avoir présenté l'état de l'art de la sécurité de l'information et les normes de la famille ISO/CEI 2700x, nouvelle référence pour les SMSI, nous exposons la démarche PDCA (Plan, Do Check, Act). En suivant cette méthode, nous établirons un état des lieux du système d'information avec le SoM (Statement of Maturity), une appréciation des risques des actifs avec le RATP (Risk Assessment Treatment Plan), un plan de continuité pour assurer une reprise des activités IT avec le DRP (Disaster Recovery Plan) et un résumé des besoins du SMSI avec le SoA (Statement of Applicability). Afin d'illustrer la mise en œuvre d'une des mesures de sécurité retenues dans le SoA, nous implémenterons une plate-forme logicielle de supervision qui permettra de réduire les risques en augmentant la qualité des services informatiques. La supervision apportera une vue globale du fonctionnement des services et hôtes spécifiés sur le LAN (Local Area Network), de la mesure de leur performance et nous alertera en cas d'incident.

Mots clés : Sécurité de l'Information, SMSI, ISO/CEI 27001, PDCA, Supervision, SNMP, Nagios, MRTG.

SUMMARY

This document is about the implementation of an ISMS (information security management system) within a SME, highly specialised in Micro-Optics. Following a state of the art on information security and the ISO/CEI 2700x standards -the new reference for ISMS- the PDCA (Plan, Do, Check, Act) approach is presented. Using this method, we will prepare a review of the information system using the SoM (Statement of Maturity); an evaluation of the assets' level of risk named : RATP (Risk Assessment Treatment Plan); a continuity plan to ensure a smooth resumption of IT activities with the DRP (Disaster Recovery Plan); and a summary of the ISMS needs using the SoA (Statement of Applicability). In order to illustrate the implementation of one of the security measure adopted in the SoA, we will implement a monitoring software platform that will enable to reduce risks by enhancing the quality of IT services. The monitoring software platform will give us an overview of the specified services and hosts on the LAN (Local Area Network) ; of a measurement of their performance and will notify us when an incident occurs.

Key words : Information Security, ISMS, ISO/CEI 27001, PDCA, Monitoring, SNMP, Nagios, MRTG.