14/02/2020

SOVILLA Flavio

SI-CA2a CPNV

**Mise en place de stockage SAN en iSCSI dans un environnement Linux**

Table des matières

[1 Analyse préliminaire 2](#_Toc33793164)

[1.1 Introduction 2](#_Toc33793165)

[1.1.1 Cadre du projet 2](#_Toc33793166)

[1.1.2 Description du projet 2](#_Toc33793167)

[1.2 Objectifs 2](#_Toc33793168)

[1.3 Planification initiale 3](#_Toc33793169)

[2 Analyse / Conception 3](#_Toc33793170)

[2.1 Concept 3](#_Toc33793171)

[2.1.1 SAN et NAS 3](#_Toc33793172)

[2.1.2 Protocole iSCSI 4](#_Toc33793173)

[2.1.3 Architecture réseau 8](#_Toc33793174)

[2.1.4 Redondance NAS 9](#_Toc33793175)

[2.1.5 Les RAID 10](#_Toc33793176)

[2.1.6 Les types de backups 12](#_Toc33793177)

[2.2 Les choix 13](#_Toc33793178)

[2.2.1 Le réseau 13](#_Toc33793179)

[2.2.2 Les backups et la redondance 14](#_Toc33793180)

[2.2.3 La sécurité 15](#_Toc33793181)

[2.3 Stratégie de test 15](#_Toc33793182)

[2.3.1 Types de tests 15](#_Toc33793183)

[2.4 Risques techniques 15](#_Toc33793184)

[2.5 Planification 16](#_Toc33793185)

[2.6 Dossier de conception 16](#_Toc33793186)

[3 Réalisation 17](#_Toc33793187)

[3.1 Dossier de réalisation 17](#_Toc33793188)

[3.2 Description des tests effectués 17](#_Toc33793189)

[3.3 Erreurs restantes 18](#_Toc33793190)

[3.4 Liste des documents fournis 18](#_Toc33793191)

[4 Conclusions 19](#_Toc33793192)

[5 Annexes 20](#_Toc33793193)

[5.1 Résumé du rapport du TPI / version succincte de la documentation 20](#_Toc33793194)

[5.2 Sources – Bibliographie 20](#_Toc33793195)

[5.3 Journal de travail 20](#_Toc33793196)

[5.4 Manuel d'Installation 20](#_Toc33793197)

[5.5 Manuel d'Utilisation 20](#_Toc33793198)

[5.6 Archives du projet 20](#_Toc33793199)

# Analyse préliminaire

## Introduction

### Cadre du projet

Le projet sera effectué au CPNV de Ste-Croix dans le cadre d’une préparation au projet TPI (Travail Professionnel Individuel) de fin d’année, qui durera environ 90h. Cette préparation se doit d’être assez proche, c’est pourquoi nous avons à disposition 12 périodes par semaine, ce qui fera un total d’environ 72h sur tout le trimestre. Nous devons notamment rédiger et signer un cahier des charges qui sera vérifier par notre chef de projet. Ce dernier nous suivra pour la totalité du projet et nous guidera si nous nous éloignons trop du chemin prévu pour le projet. Nous devons aussi rendre des comptes rendus à la fin des différents sprints ainsi qu’une documentation complète. Comme vous pouvez le lire, ce projet se rapproche de notre TPI de fin d’année à l’exception des experts et de quelques heures en moins.

### Description du projet

Afin de ne pas empiéter sur le projet de fin d’année, j’ai demandé à mon chef de projet de me donner un sujet différent mais qui est en rapport avec le choix que j’avais fait. Dès lors, je me suis vu attribué la tâche de :

« Mettre en place un stockage SAN en iSCSI dans un environnement Linux ».

Cette dernière a pour but d’approfondir mon savoir avec un environnement Linux, tout en gardant un côté réseau et sécurité. En effet, le but sera de créer un réseau SAN avec deux NAS et d’en assurer la sécurité des données avec un système de redondance. De plus, je devrais utiliser un nouveau protocole encore jamais vu, le protocole iSCSI. Nous en avions déjà entendu parlé mais nous n’avons jamais vu de quoi il s’agissait. Enfin, il faudra créer une infrastructure sécurisée pour l’environnement du SAN.

La méthode AGILE sera utilisée dans le cadre de ce projet. Des sprints seront créés avec une date butoir. A chaque fin de sprint une entrevue avec le chef de projet se doit d’être faite pour évaluer la situation et apporté en temps réel des modifications sur le projet. Ainsi selon les problèmes ou solutions rencontrés, le planning avec les sprints sera modifié au fur et à mesure de la progression.

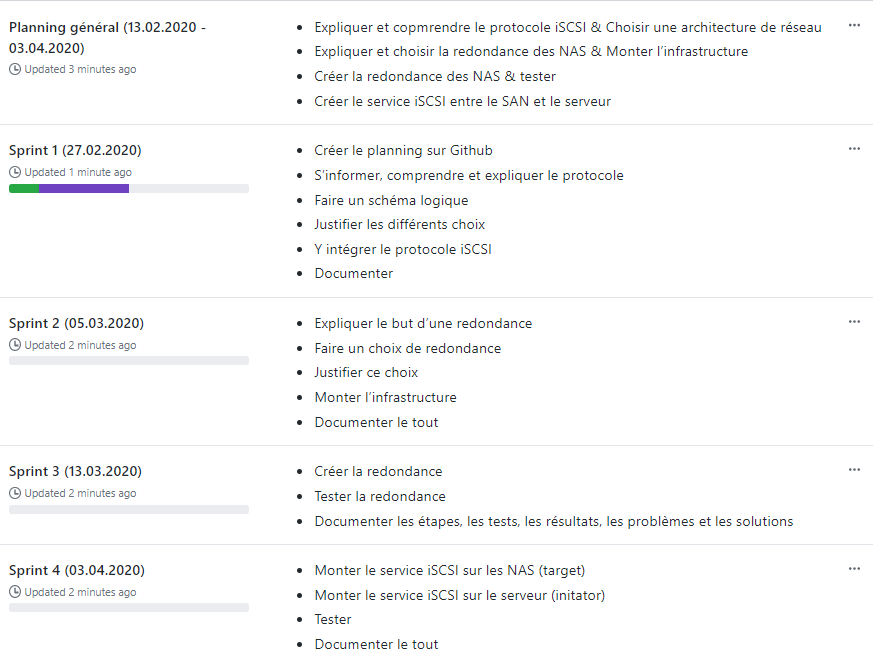
## Objectifs

*Ce chapitre énumère les objectifs du projet. L'atteinte ou non de ceux-ci devra pouvoir être contrôlée à la fin du projet. Les objectifs pourront éventuellement être revus après l'analyse.*

*Ces éléments peuvent être repris des spécifications de départ.*

## Planification initiale

Voici le planning initial généré sur Github :



Il est fort probable que ce planning subisse des modifications. En effet, nous ne pouvons savoir à l’avance si certaines tâches sont assez précises concernant le temps consacré. Certaines pourront être faites plus rapidement et d’autres prendront peut-être plus de temps. Ce planning subira donc des modifications au fur et à mesure de l’avancement du projet. Les problèmes rencontrés, les solutions pour y remédier ainsi que la modification du planning seront expliqués plus bas lorsque cela sera nécessaire.

# Analyse / Conception

## Concept

### SAN et NAS

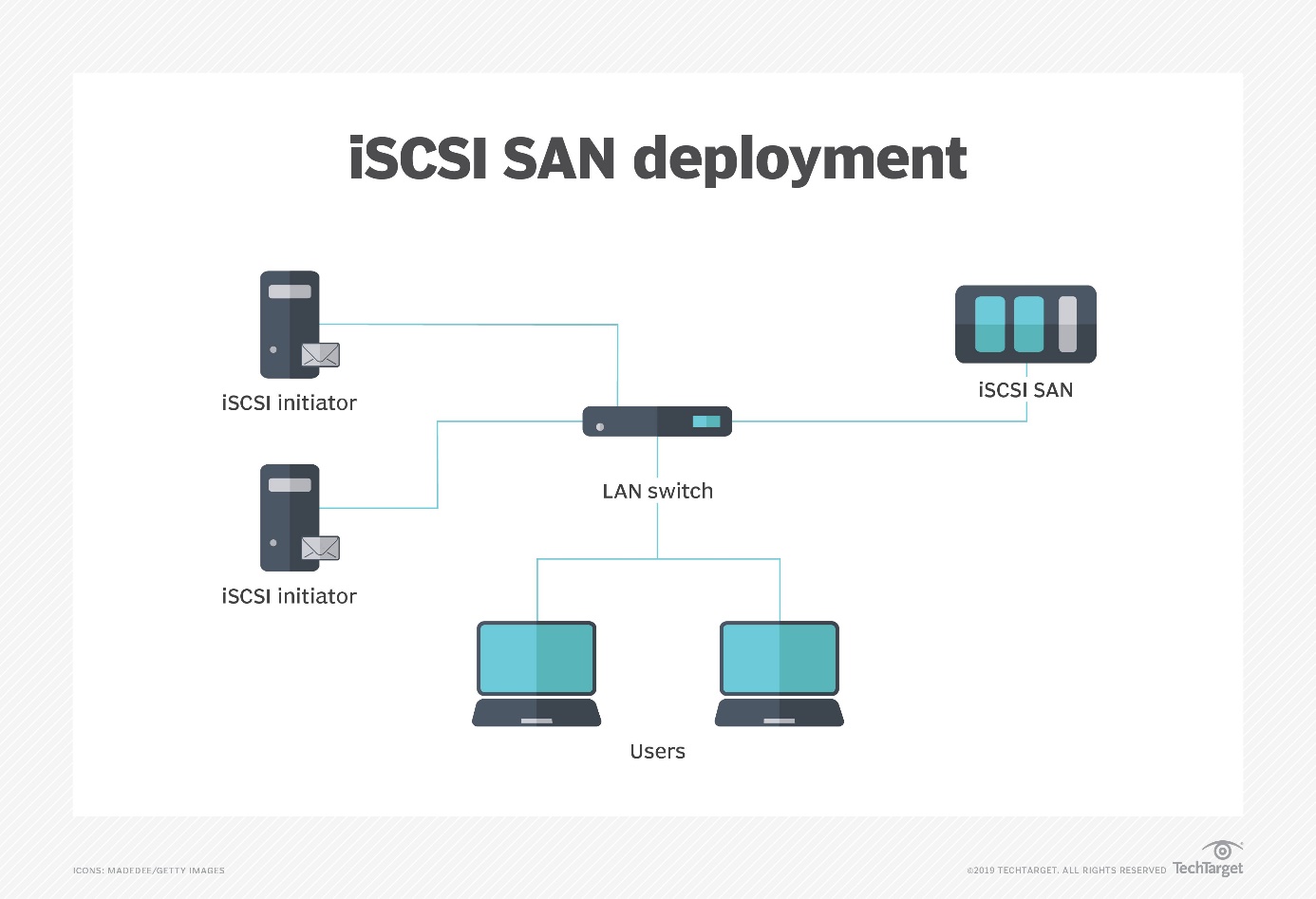
Tout d’abord, avant de se lancer sur ce qu’est le protocole iSCSI, il va falloir comprendre dans quel but nous allons l’utiliser.

Dans notre petit réseau, nous voulons créer ce qui s’appelle un SAN ou Storage Area Network. Ce SAN, sera créé avec l’aide de deux NAS ou Network Attached Storage. Ce sont deux solutions de stockage en réseau, cependant elles diffèrent quant à l’utilisation. En effet, alors que le NAS va offrir un stockage sur de simples fichiers le SAN, lui, va offrir un stockage sur un bloc de disque. Les protocoles sont notamment différents, le SAN va utiliser SCSI, Fibre Channel ou SATA alors que le NAS va utiliser un Serveur de fichiers, NFS ou CIFS. A noter qu’une infrastructure SAN sera plus coûteuse et plus complexe à installer contrairement à son homologue, un NAS, dont l’infrastructure sera moins coûteuse et complexe.

Nous n’allons pas plus nous étendre sur la technologie NAS. Par contre, comme nous allons utiliser la technologie SAN, voici de plus amples explications.

Comme vous avez pu le lire, SAN utilise SCSI. Ce protocole a été amélioré pour donner deux protocoles plus puissants : Fibre Channel et iSCSI (que nous expliquerons plus tard). Ces derniers permettent des débits plus puissants et sur de plus longues distances. Nous pourrons alors connecter plus d’appareils sur le SAN et ils auront tous accès aux différents disques.

Voici un schéma plus explicatif :



Comme vous pouvez le voir, nous retrouvons notre SAN relié à un switch, lui-même relié à des utilisateurs et des iSCSI initiateurs. Ici, notre SAN n’est pas protégé, mais il sert d’espace disque de stockage où tout le monde à accès.

### Protocole iSCSI

**Comment fonctionne-t-il**

Le protocole iSCSI ou internet Small Computer System Interface est un protocole initié en 1990 et lancé sur le marché en 1998. Il a pour but de relier des installations de stockage de données. Il va faciliter le transport des données sur un LAN ou un WAN en utilisant les réseaux IP sur de courtes ou longues distances.

Le fonctionnement est très simple, le client (appelé initiateur), va envoyer une commande SCSI sur des périphériques de stockage utilisant iSCSI (appelés cibles) qui peuvent se trouver en local ou sur des serveurs distants (appelés SAN), qui rassemblent les ressources de stockages en un point unique et offre un disque atteignable pour les clients contenant des fichiers. Une fois que la commande a atteint la cible, cette dernière va la récupérer, la lire, récupérer les informations et pour finir envoyer une réponse à l’initiateur.

Dans le cadre de notre projet, nous allons créer un SAN sur un LAN. Il ne sera atteignable que sur le réseau et non pas l’extérieur (cf. WAN). Un schéma logique du réseau sera présenté plus tard.

**L’encapsulation**

Voici une image montrant une encapsulation d’un fragment iSCSI :



On constate donc que c’est comme une encapsulation commune du modèle TCP/IP, à l’exception d’iSCSI qui vient se loger avant le segment TCP de la couche protocole (ici, la couche application avec les données est omise).

Dans l’ordre d’encapsulation, nous avons :

- iSCSI qui va fournir les instructions d’ordre et de control avec les commandes SCSI et si besoin est, des données supplémentaires.

- iSCSI mis dans un segment avec le protocole TCP (Transmission Control Protocol) qui va permettre un envoi sûr du segment grâce à son protocole de vérification.

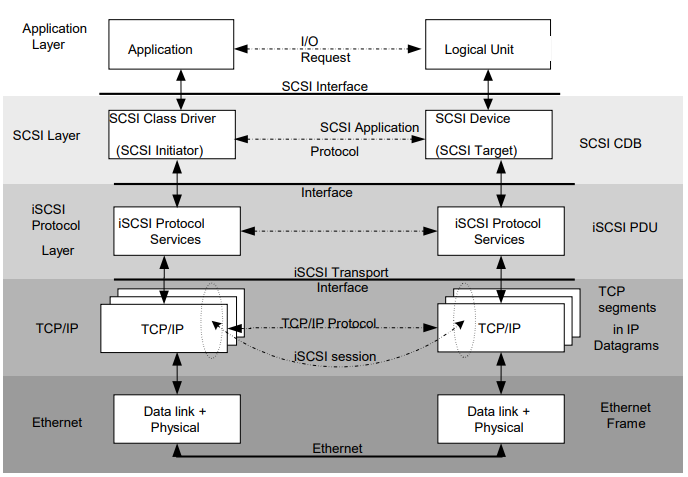
- Le segment est ensuite englobé dans un paquet IP, qui va permettre les échanges à distance avec d’autres protocoles à l’intérieur, notamment la possibilité d’une fragmentation du paquet si le flux doit être restreint ou encore le Time to Live pour que le paquet n’encombre pas le réseau.

- Et pour finir le tout s’encapsule dans une trame Ethernet, qui elle, contiendra les adresses MAC pour l’échange de données sur un LAN. A noter que cette trame contient un en-tête avec les informations habituelles comme le préambule, le SFD (Start of Frame Delimiter), … et une queue de bande (ou FCS, Frame Check Sequence) qui permet la vérification de la trame.

Une fois tout cela encapsulé, la trame est prête à être envoyée en bit sur le réseau physique.

**Les couches**

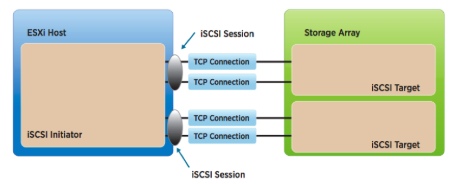
Voici une image montrant les différentes couches d’iSCSI :



Comme vous pouvez le voir, ici nous retrouvons notre encapsulation pas à pas, mais cette fois, avec une vue des différentes couches du protocole TCP/IP. Il y a notamment la couche application qui a été omise lors de l’explication du sous chapitre précèdent.

Le petit plus à noter est le SCSI Layer avec le SCSI CDB (Commande description Block) qui se joint entre le protocole iSCSI (qui va encapsuler la commande) et la couche application qui va permettre de donner l’ordre à la couche SCSI. En effet, une fois que nous avons souhaité faire une demande aux disques de notre SAN, SCSI Layer va transformer cette demande afin de garantir l’interopérabilité entre les disques des différents fournisseurs et ainsi garantir un accès quel que soit le support (les commandes SCSI ont été normalisée dans ce but, pour de plus amples informations vous pouvez rechercher la norme ANSI INCITS 408-2005).

Il existe notamment une session iSCSI sur la couche TCP/IP, cette dernière sert à garantir l’échange entre l’initiateur et la cible. Comme mentionné plus haut, iSCSI utilise le protocole TCP pour initier cette session avec un SYN/ACK et va utiliser une iSCSI session ID (ISID) pour l’identification. Il est possible d’avoir de multiples sessions ouvertes entre l’initiateur et la cible comme présenté dans l’image suivante :



Il existe notamment une Connection ID (CID) qui sert à identifier à l’aide d’un ID de connexion (tout est dans le nom) et un Target Session Identifying Handle (TSIH) qui permet de cibler une cible spécifique pour une session avec un nom d’initiateur.

De notre côté, pour présenter les disques aux clients, nous allons utiliser ce qui s’appelle une LUN (Logical Unit Number), qui va nous permettre de mettre un numéro d’identification à notre unité de stockage. Une fois ce numéro attribué, nous avons la possibilité de gérer si nous souhaitons masquer ou non l’espace de stockage à un serveur en particulier (cela s’appelle le LUN Masking). Mais comme nous ne possédons qu’un serveur et dans le but de simplifier l’infrastructure, nous allons lui attribuer une connexion avec notre numéro de LUN afin qu’il puisse communiquer avec.

**La sécurité**

Il existe plusieurs méthodes pour sécuriser l’échange d’informations et d’en garantir la sécurité. Nous n’utiliserons bien entendu pas toutes les méthodes car certaines se révèlent être complexes et hors de portée, mais en utilisant les principales, nous serons capables d’offrir des connexions sécurisées à notre SAN.

Voici les méthodes que nous utiliserons :

* **CHAP authentification** : (Challenge-Handshake Authentification Protocol) Permet une authentification pour se connecter à la cible. Il existe aussi le Mutual CHAP qui va exiger une authentification des deux côtés et qui est plus sécurisé.
* **Autorisation** : Une gestion des autorisations en fonction des personnes authentifiées. Ce qu’elles peuvent lire, modifier ou exécuter.
* **Encryptions** : Ajouter un clé d’encryptions (AES-256) sur les dossiers partagés pour en protéger le contenu (attention les clés doivent être gardées).

Quelques autres méthodes qui permettent une sécurité plus performante mais qui demande un travail plus fourni (je n’explique que la fonction principale) :

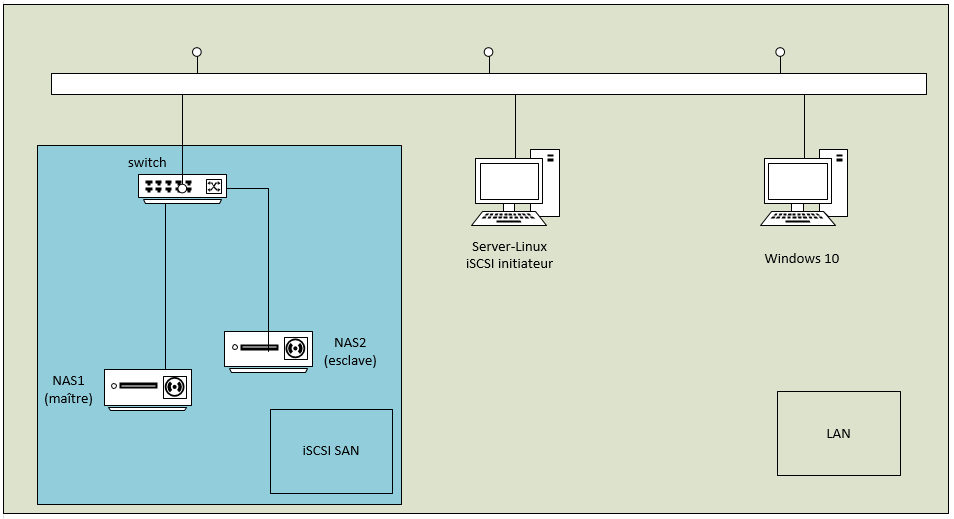
* **RADIUS Authentification** : (Remote Authentification Dial-In user Service) Permet un accès à un serveur pour garantir l’authentification.
* **IPSec Authentification** : Comme le CHAP authentification n’est pas assez sécurisé, ce protocole offre une couche de sécurité supplémentaire sur le paquet de la couche IP.
* **IPSec Encrpytion** : Pour fortifier l’encryptions des données.

### Architecture réseau

Il est possible de créer différentes architectures du réseau. Cependant, nous ne verrons que deux architectures et nous choisirons la plus efficace et complète.

**Réseau en BUS**

Une possibilité est de monter un réseau en BUS comme ci-dessous.

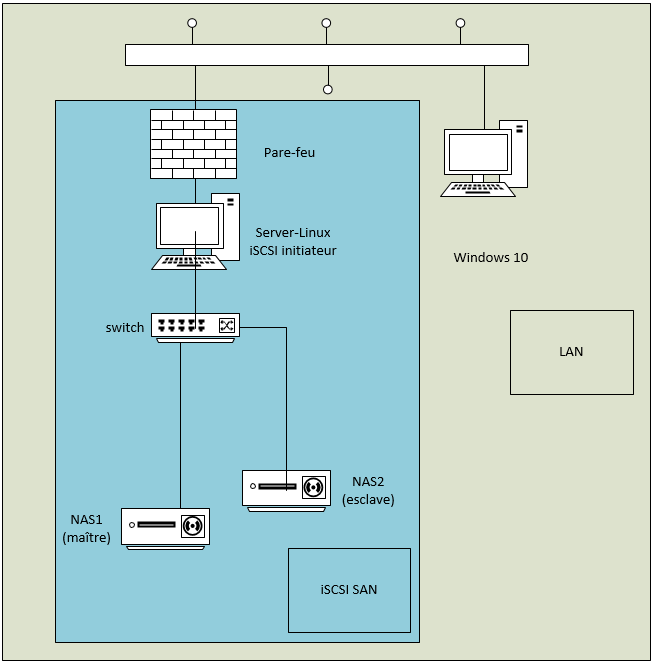


Nous constatons qu’il existe un LAN avec un SAN sur le même réseau. Le SAN est donc ouvert et à portée de tous les utilisateurs sur le réseau. Ce n’est donc pas une bonne idée. En effet, il se peut qu’une personne mal intentionnée ou ne faisant pas exprès perturbe le fonctionnement du SAN et de ce fait le rende inatteignable. La disponibilité et l’intégrité des données du NAS sont donc en « danger ». C’est pourquoi cette disposition n’est pas préconisée.

**Réseau Hybride**

Contrairement à un simple réseau en BUS, nous allons créer un réseau hybride. Ce dernier sera composé d’une partie en BUS et d’une autre en étoile.

Voici ce que à quoi va ressembler le réseau :



Nous retrouvons ici notre topologie en bus pour le LAN, mais, avec cette fois, une topologie en étoile pour le SAN. Cette disposition, va permettre une plus grande sécurité des données du SAN. Comme vous pouvez le voir, le SAN sera protégé par un pare-feu qui sera capable de gérer les entrées et les sorties avec des règles et ainsi en garantir une plus ample protection.

### Redondance NAS

**Le principe**

Comme vous avez pu le constater, j’ai exprès omis de vous parler des deux NAS dans l’infrastructure du SAN. C’était voulu. En effet, bien qu’à eux deux ils composent le réseau SAN, ils vont servir à notamment créer la redondance. Mais qu’est-ce qu’une redondance ? Le Larousse nous offre une définition plus ou moins exhaustive de ce terme : « En informatique et dans les télécommunications, duplication d'informations afin de garantir leur sécurité en cas d'incident ». Cela paraît presque complet, mais pas tout à fait. Voici une explication plus exhaustive : différents problèmes peuvent survenir sur une machine, une surtension qui provoque un court-circuit, un disque dur usé, une mauvaise manipulation qui efface des dossiers importants, etc… C’est pourquoi en informatique, nous effectuons ce qui s’appelle un backup des informations afin de garantir la sécurité des informations qui sont sur les disques durs. Ce backup se doit d’être redondant afin de perdre le moins de données possibles. C’est-à-dire qu’il faut faire des backups sur des intervalles de temps courts. Grâce à cela, s’il faut récupérer une donnée, il est possible de le faire presque en l’état.

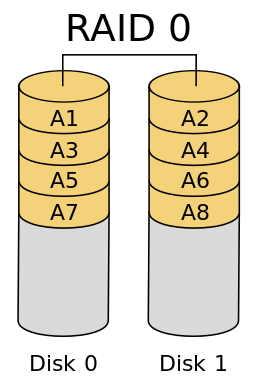
Nous allons donc créer une redondance avec les deux NAS. L’un, dit maître, sera celui sur lequel les informations sont directement stockées. L’autre, dit esclave, servira à enregistrer les backups du maître et en garantir la sécurité avec une redondance. Ainsi si un problème survient sur le maître, nous possédons une réplique de toutes les informations nécessaires sur l’esclave. Et l’inverse est vrai aussi, si l’esclave a un problème, nous pouvons tout retransférer sur ce dernier depuis le maître.

### Les RAID

La dernière phrase du paragraphe précèdent n’est pas totalement juste. En effet, il faudrait préconiser les problèmes sur l’esclave car il contient tous les backups. Il est donc important de protéger les données sur ses disques (nous utiliserons aussi cette technique sur les disques du maître pour en accroître la sécurité). Pour cela, nous allons faire appel à ce qui s’appelle du RAID. Il en existe différents types : RAID 0, 1, 5, 6 et 10 sont les plus connus. Je vais ici m’attarder que sur le RAID 0, 1 et 5. C’est ce qui nous intéresse le plus. Voici donc une petite explication de ces derniers.

**RAID 0**

Aussi appelé « volume agrégé par bandes », sa fonctionnalité consiste à augmenter le les performances de la grappe (comme plusieurs disques sont utilisés, nous parlons de grappe) car on va utiliser plusieurs disques simultanément. Voici un exemple en image :

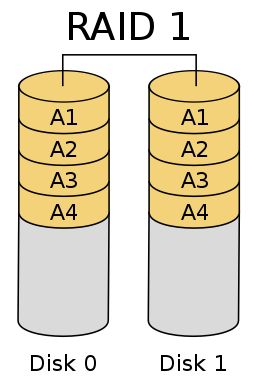


Comme on peut le voir, les informations se divisent sur deux différents disques, formant une grappe, et si nous souhaitons écrire ou lire des informations, ces actions seront faites deux fois plus rapidement. Le gros problème avec cette disposition est qu’il suffit qu’un seul des disques rencontre un problème et ne peut plus être utilisé et c’est la totalité des données qui est perdu. Il est notamment possible d’utiliser plus que deux disques pour créer cette grappe, cependant un minimum de deux est requis.

Concernant l’espace de stockage, celui-ci est donc ‘agrandit’. En effet, si nous avons deux disques de 1To, nous aurons 2To d’espace libre pour stocker nos données.

**RAID 1**

Aussi appelé « miroir », sa fonctionnalité consiste à dupliquer toutes les données. Celles-ci seront alors sécurisées sur deux disques différents. Voici un exemple en image :

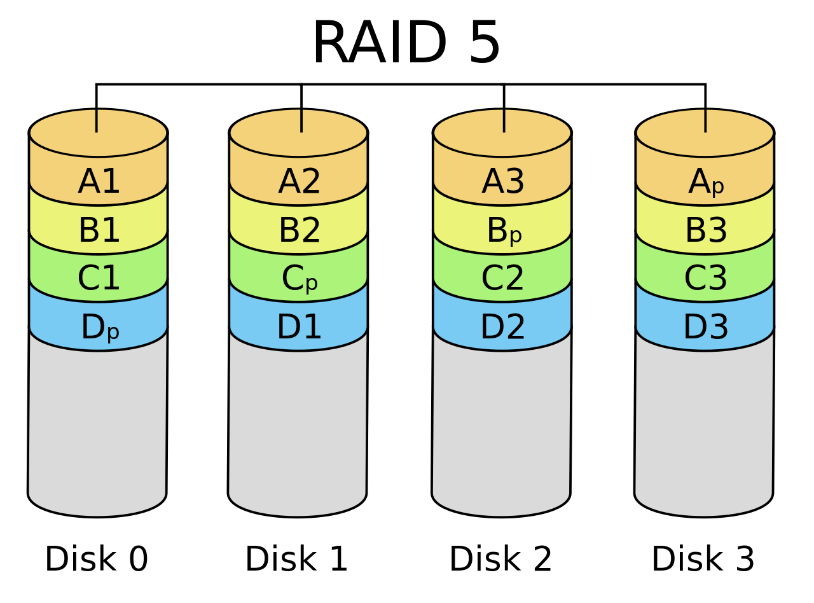


Comme on peut le constater, les informations sont dupliquées et donc, contrairement au RAID 0, elles sont sécurisées en cas de perte d’un des deux disques. Ici, un minimum de deux disques se doit d’être utilisé.

Ici, les données seront ‘divisées’ comme nous utilisons un disque pour copier le tout. Si nous avons par exemple deux disques de 1To, alors nous n’aurons que 1To d’espace de stockage de données.

**RAID 5**

Aussi appelé « volume agrégé par bandes à parité répartie ». Cette grappe va utiliser la méthode du RAID 0 et y ajouté une parité. Voici une image explicative :



Comme on peut le voir sur la première tranche, nous retrouvons la même disposition que le RAID 0, cependant, une parité est ajoutée. Cette dernière va nous servir à reconstruire le tout si un disque est perdu. Ainsi si nous perdons un disque il est possible de tout reconstruire. Cependant, nous ne pouvons perdre qu’un unique disque. En effet, perdre deux disque signifierais la perte de toutes les données. Ici, nous avons une image avec 4 disques mais un minimum de 3 disques est requis pour effectuer ce type de grappe. Pour la vitesse d’écriture, celle-ci est un peu accrue comme pour le RAID 0 mais elle va perdre un peu de sa vitesse à cause de la parité. La vitesse de lecture dépendra de la répartition sur les disques mais elle sera accrue.

Supposons que nous possédons le minimum requis de 3 disques de 1To chacun, alors nous ne posséderons qu’un espace de 2To de données. La parité prendra sera répartie sur les disques mais prendra un tiers des données.

### Les types de backups

Il existe trois types de backups. Ils sont tous des fonctionnalités différentes et peuvent être utilisés selon divers besoins.

**Complet**

Comme son nom l’indique, c’est une sauvegarde complète des données. Elle est donc à utiliser le moins possible. En effet, selon la quantité de données à importer elle peut s’avérer longue. Elle est donc à effectuer lorsque personne ne manipule les données et à ne pas faire tous les jours. Il est conseillé d’en faire une toutes les semaines à un moment où personne ne travaille (durant la nuit du weekend par exemple).

**Différentiel**

C’est un type de sauvegarde un peu plus complexe. Mais elle va nous permettre de gagner du temps et de l’espace de stockage. Cette technique consiste à ne sauvegarder que les derniers fichiers modifiés depuis le dernier backup complet. On utilisera ensuite ce dernier ainsi que la sauvegarde différentielle pour recrée une sauvegarde complète du jour J avec tous les fichiers modifiés. Nous allons donc à chaque fois faire un backup des nouvelles données en fonction de la sauvegarde complète. Il en adviendra que les données seront de plus en plus ‘lourdes’ au fil des jours. Cependant il ne faudra qu’une sauvegarde différentielle et la complète pour retrouver toutes les données.

**Incrémentiel**

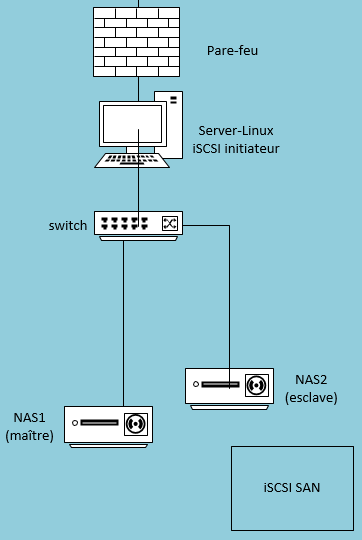
Ici, nous avons la technique la plus légère. Cette technique va aussi demander une sauvegarde complète et à partir de celle-ci, faire des backups réguliers. Elle ne prendra en compte que les fichiers modifiés, mais au lieu de prendre référence sur la complète, elle va regarder son ancienne sauvegarde incrémentale. Ainsi elle n’augmentera pas au fil des jours. Le désavantage est qu’il faudra récupérer toutes les données incrémentielles précédentes pour faire une restauration complète. Cependant comme elle prend moins de données en compte, elle nous permettra de n’utiliser que peu de flux pour la sauvegarde et ainsi permettre un backup sur un court instant pour permettre de retrouver les données perdues sur un temps court.

## Les choix

### Le réseau

Pour la construction de notre réseau, nous allons choisir le réseau hybride afin de garantir un minimum de sécurité pour le NAS. Cette topologie est certes un peu plus compliquée à mettre en place que le réseau en BUS mais elle nous permettra d’étendre nos connaissances sur le déploiement d’un réseau plus sécurisé ainsi que nous améliorer dans le domaine du système.

Voici ce à quoi va ressembler le réseau :



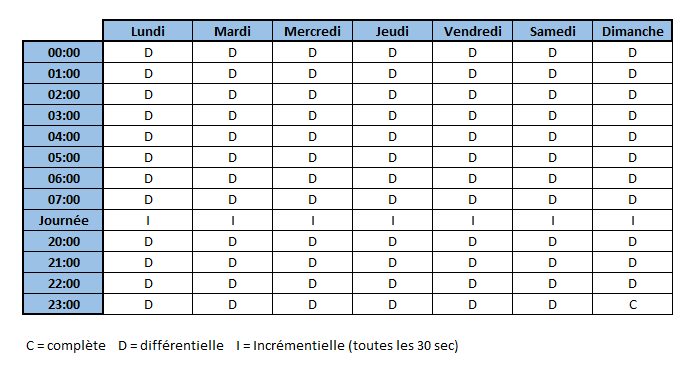
Nous aurons toujours la possibilité de l’agrandir, ajouter des ressources et logiciels ainsi que de tester le tout si nous parvenons à tout finir en avance.

### Les backups et la redondance

Comme nous possédons quatre disques sur chaque NAS et qu’il est très peu probable que 2 disques cassent en même temps, il serait bien d’opter pour le RAID 5 afin de garantir tout de même une sécurité pour les données mais aussi augmenter la vitesse de lecture et d’écriture sur ceux-ci.

Supposons maintenant que nous n’avons vraiment pas de chance et que deux disques deviennent soudainement défectueux, nous pouvons compter sur notre NAS esclave qui garde nos données bien au chaud. Pour cela une sauvegarde complète (supposons un dimanche soir lorsqu’il y a le plus de chance de personne de travail) suivi d’incrémentielles toutes les 30 secondes pour garantir une récupération immédiate durant la journée et une différentielle le soir ainsi que toutes les heures pendant la nuit (car peu de personnes travaillent de nuit dans une PME).

Voici un petit tableau qui résumerait les sauvegardes :



### La sécurité

Comme expliqué plus haut, nous n’allons utiliser que les bases de la sécurité pour ce projet. Elles consisteront au CHAP Authentification, au cryptage des données et à l’autorisation des utilisateurs. Comme le tout, il sera possible de modifier et d’en ajouter au fur et à mesure du projet, mais elles resteront les principales sécurités.

Concernant la sécurité d’accès, il est possible de mettre un pare-feu sur le serveur pour augmenter la sécurité, cependant il est fort probable que le temps ne le permette pas, mais tout dépendra de l’avancement du projet.

## Stratégie de test

*Décrire la stratégie globale de test:*

* *types de des tests et ordre dans lequel ils seront effectués.*
* *les moyens à mettre en œuvre.*
* *couverture des tests (tests exhaustifs ou non, si non, pourquoi ?).*
* *données de test à prévoir (données réelles ?).*
* *les testeurs extérieurs éventuels.*

### Types de tests

## Risques techniques

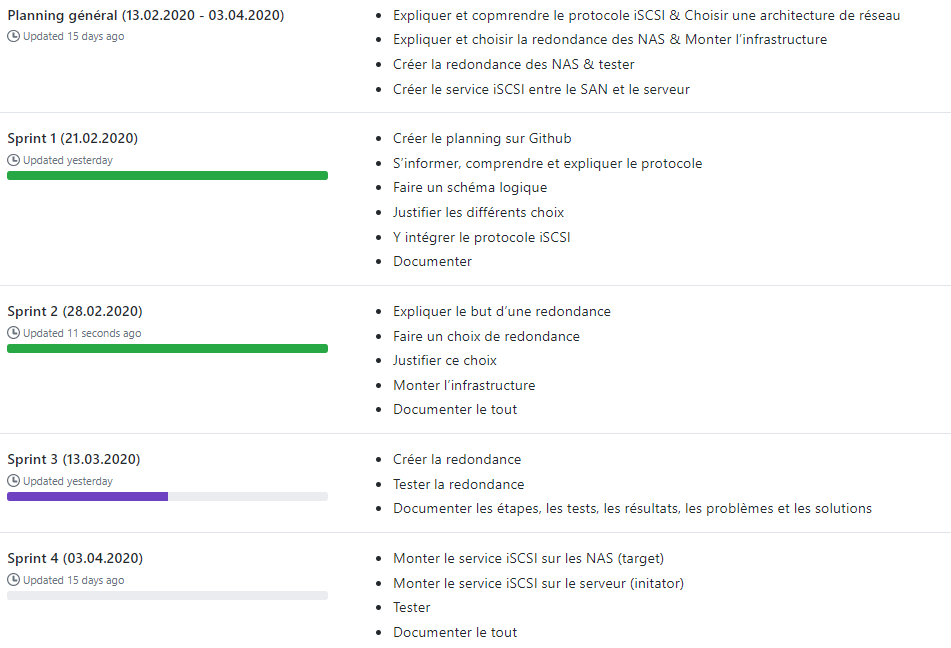
Les premiers risquent techniques viennent d’une part du manque de compétences. En effet, n’ayant jamais rien fait avec Linux à part des lignes de commandes pour créer des dossiers ou gérer des droits, il est flagrant qu’une absence de connaissances dans cet OS peut être problématique. À ajouter que c’est la première fois que je rencontre le protocole iSCSI et que je n’en avais jamais entendu parler auparavant.

Mes seules connaissances sont celles de la redondance des sauvegardes pour la sécurité que nous avions déjà un peu travaillée au préalable.

Aucunes actions n’ont été entreprises pour combler ces lacunes. Le but étant de se lancer dans un nouvel environnement et d’essayer de prouver mon indépendance face à une problématique inconnue. Ainsi cela me permettra d’améliorer mes capacités en tant qu’informaticien et développé mes compétences en autodidacte.

## Planification

Voici une modification du planning. Les recherchent ayant pris moins de temps que prévu, j’ai avancé les dates des premiers sprints (dates auxquelles j’ai fini) et j’ai laissé les autres dates afin de laisser une plus grande marge, mais peut-être que se sera terminé avant.



## Dossier de conception

*Fournir tous les document de conception:*

* *le choix du matériel HW*
* *le choix des systèmes d'exploitation pour la réalisation et l'utilisation*
* *le choix des outils logiciels pour la réalisation et l'utilisation*
* *site web: réaliser les maquettes avec un logiciel, décrire toutes les animations sur papier, définir les mots-clés, choisir une formule d'hébergement, définir la méthode de mise à jour, …*
* *bases de données: décrire le modèle relationnel, le contenu détaillé des tables (caractéristiques de chaque champs) et les requêtes.*
* *programmation et scripts: organigramme, architecture du programme, découpage modulaire, entrées-sorties des modules, pseudo-code / structogramme…*

***Le dossier de conception devrait permettre de sous-traiter la réalisation du projet !***

Pour pouvoir mener à bien ce projet, nous allons avoir besoin du matériel suivant :

* 1x Synology NAS DS413j possédant 4 disques HDD de 1Tbits (4Tbits au total)
* 1x Synology NAS DS413j possédant 4 disques HDD de 500Mbits (2Tbits au total)
* 2x Prises pour alimenter les NAS DS413j
* 2x Switch Netgear GS105 v4 de 5 ports 10/100Mbits (avec alimentation)
* 3x câble RJ-45 pour les connexions
* 1x Ordinateur tour avec un serveur Ubuntu 18.04

# Réalisation

## Dossier de réalisation

*Décrire la réalisation "physique" de votre projet*

* *les répertoires où le logiciel est installé*
* *la liste de tous les fichiers et une rapide description de leur contenu (des noms qui parlent !)*
* *les versions des systèmes d'exploitation et des outils logiciels*
* *la description exacte du matériel*
* *le numéro de version de votre produit !*
* *programmation et scripts: librairies externes, dictionnaire des données, reconstruction du logiciel - cible à partir des sources.*

*NOTE : Evitez d’inclure les listings des sources, à moins que vous ne désiriez en expliquer une partie vous paraissant importante. Dans ce cas n’incluez que cette partie…*

## Description des tests effectués

*Pour chaque partie testée de votre projet, il faut décrire:*

* *les conditions exactes de chaque test*
* *les preuves de test (papier ou fichier)*
* *tests sans preuve: fournir au moins une description*

Le premier test effectué est le matériel. Il faut tester que tout fonctionne. Il suffit donc d’allumer l’ordinateur, le switch et les Synology. Tout était fonctionnel et je n’ai remarqué aucun problème.

Une fois que le tout est prêt et opérationnel, on peut tester s’ils se voient sur le réseau. Pour ma part, j’ai installé l’assistant NAS qui permet de scanner le réseau et de trouver les différents NAS relié. J’ai fait cela avec ma machine sous Windows pour voir s’il n’y avait pas de problèmes. Ensuite j’ai fait le test avec ma machine virtuelle Linux mais cela n’a donné aucun résultat. J’ai donc commencé par mettre la VM (Virtual Machine) en mode bridge et non NAT (Network Address Translation) afin qu’elles se partagent la même carte réseau et donc l’IP. A nouveau je ne pouvais pas atteindre les NAS depuis la VM mais depuis Windows oui. J’avais déjà eu un problème de ce genre lors d’un précèdent projet où je devais installer Pfsense pour gérer les authentifications sur un réseau et malgré l’aide de Mr. Jaggi un mois durant, nous ne sommes pas arrivés à faire passer ne serait-ce qu’un simple ping de la VM en dehors de la machine physique. Nous avions conclu à un problème avec VMware. Afin de ne pas perdre trop de temps avec ce problème, j’ai donc demandé une tour pour installer Linux directement dessus. Une fois cela fait, je me suis rendu compte que depuis plus d’un mois nous avions un problème et qu’il était impossible de se connecter à un internet avec ce câble. Il est possible d’atteindre le réseau, les autres ordinateurs et la Gateway mais pas en sortir. Je vais donc prendre le câble de mon poste pour le relier à internet lorsqu’il faudra faire des mises à jours ou télécharger différents logiciels.

## Erreurs restantes

*S'il reste encore des erreurs:*

* *Description détaillée*
* *Conséquences sur l'utilisation du produit*
* *Actions envisagées ou possibles*

## Liste des documents fournis

*Lister les documents fournis au client avec votre produit, en indiquant les numéros de versions*

* *le rapport de projet*
* *le manuel d'Installation (en annexe)*
* *le manuel d'Utilisation avec des exemples graphiques (en annexe)*
* *autres…*

# Conclusions

*Développez en tous cas les points suivants:*

* *Objectifs atteints / non-atteints*
* *Points positifs / négatifs*
* *Difficultés particulières*
* *Suites possibles pour le projet (évolutions & améliorations)*

# Annexes

## Résumé du rapport du TPI / version succincte de la documentation

## Sources – Bibliographie

*Liste des livres utilisés (Titre, auteur, date), des sites Internet (URL) consultés, des articles (Revue, date, titre, auteur)… Et de toutes les aides externes (noms)*

## Journal de travail

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Durée** | **Activité** | **Remarques** |
|  |  |  |  |
|  |  |  |  |

## 

## Manuel d'Installation

## Manuel d'Utilisation

## Archives du projet

*Media, … dans une fourre en plastique*