

Second rendu SAÉ 2.03

Rapport technique des solutions logicielles

BAERT Thomas, CERISIER Robin, ROUBAUD Louis



Table des matières

1	Introduction	3
1.1	Contexte	3
1.2	Problématique	3
2	Pré-requis	3
3	Solutions logicielles	3
3.1	Base de données	3
3.1.1	Protocoles et normes associés aux bases de données	3
3.1.2	PostgreSQL	4
3.1.3	Oracle Database	5
3.1.4	MySQL	5
3.1.5	Étude comparative	5
3.2	Serveur web	7
3.2.1	Protocoles et normes associés aux serveurs web	7
3.2.2	Apache	7
3.2.3	Nginx	8
3.2.4	Étude comparative	8
3.3	Service de messagerie	9
3.3.1	Protocoles et normes associés aux services de messagerie	9
3.3.2	Postfix	9
3.3.3	Exim	10
3.3.4	Étude comparative	10
4	Conclusion	11

1 Introduction

1.1 Contexte

Dans le cadre de la SAÉ 2.03, nous avons été amenés à réaliser un schéma de réseau informatique pour une entreprise fictive. Afin de compléter ce schéma, nous allons vous présenter des solutions logicielles adaptées à l'entreprise et ces problématiques.

1.2 Problématique

L'entreprise fictive a pour but d'offrir un service de coworking à d'autres entreprises. Dans ce cadre, elle doit être capable de stocker des données de manière sécurisée, avec redondance. Elle doit également être capable de gérer des pages webs statiques et dynamiques. De plus l'entreprise a décidé de mettre en place un service de messagerie.

2 Pré-requis

Les solutions logicielles doivent répondre aux besoins suivants :

- Stockage de données avec redondance
- Gestion de pages webs statiques et dynamiques
- Service de messagerie
- Accessibilité de tout le réseau sans manipulations particulières
- Sécurisation des données

3 Solutions logicielles

3.1 Base de données

Dans cette section, nous allons vous présenter les solutions disponibles pour le stockage de données et notre recommandation.

3.1.1 Protocoles et normes associés aux bases de données

Les bases de données relationnelles ou non relationnelles utilisent différents protocoles pour le transport de données, la structuration des requêtes, ou encore l'authentification et la sécurité.

Protocole/Norme	Description
SQL (ISO/IEC 9075)	<i>Structured Query Language</i> : langage normalisé pour la gestion des bases de données relationnelles. Permet la création, la manipulation et la requête de données via des instructions textuelles.
ODBC (Open Database Connectivity)	Interface standardisée permettant aux applications d'accéder à des SGBD via un pilote. Utilisé principalement dans des environnements hétérogènes ou Windows.
JDBC (Java Database Connectivity)	Interface Java standardisée similaire à ODBC, permettant la connexion à une base de données depuis une application Java.
Protocole natif (MySQL, PostgreSQL, etc.)	Chaque moteur de base de données implémente son propre protocole binaire ou textuel pour l'échange de requêtes et de résultats entre client et serveur. Exemple : le protocole client/serveur de PostgreSQL ou MySQL.

TABLE 1 – Principaux protocoles et interfaces utilisés pour les bases de données

3.1.2 PostgreSQL

Propriété	Valeur
Éditeur	PostgreSQL Global Development Group
Brève description technique	SGBD relationnel open source, robuste et sécurisé
Licence	Open source (similaire à MIT/BSD)
Qualité documentation	Très complète et unifiée
Popularité	DBMS of the Year (DB-Engines 2023).
Taux d'utilisation	Utilisé par 49% des développeurs (stackoverflow 2024)
Avis de la communauté	Très bonne réputation, très utilisé dans le monde professionnel

TABLE 2 – Caractéristiques principales de PostgreSQL

3.1.3 Oracle Database

Propriété	Valeur
Éditeur	Oracle
Brève description technique	SGBD relationnel propriétaire, conçu pour les applications critique
Licence	Licence commerciale propriétaire
Qualité documentation	Très complète et professionnelle mais complexe
Popularité	Premier en indice de popularité. (Wikipedia)
Taux d'utilisation	Plus de 33 Md de requêtes par jour estimées (Enterprise App Today)
Avis de la communauté	Très robuste mais considéré comme très cher, avec des audits très agressifs d'Oracle

TABLE 3 – Caractéristiques principales de Oracle Database

3.1.4 MySQL

Propriété	Valeur
Éditeur	MySQL AB
Brève description technique	SGBD relationnel open-source polyvalents. Nécessite InnoDB pour les transactions
Licence	Licence public générale GNU, licence Oracle pour les entreprises
Qualité documentation	Claire et bien structuré, mais avec des répétitions et des explications pas toujours pédagogique
Popularité	Premier SGBD relationnel mondial (42% de parts de marchés)
Taux d'utilisation	39% des sites webs l'utilise. Utilisé par Apple, Amazon
Avis de la communauté	Simple à déployer et prendre en main, performant, écosystème riche

TABLE 4 – Caractéristiques principales de MySQL

3.1.5 Étude comparative

Ces trois SGBD peuvent être pertinents dans le cadre de notre projet. Cependant, nous avons décidé de choisir **PostgreSQL** pour les raisons suivantes :

- Il est open-source et gratuit, ce qui est un avantage pour une jeune entreprise.
- Il est très populaire et utilisé par de nombreuses entreprises, ce qui garantit une bonne communauté de développeurs.
- Il est robuste et sécurisé, ce qui est essentiel pour le stockage de données sensibles.
- Il est compatible avec de nombreux systèmes d'exploitation et langages de programmation.

Tandis que Oracle Database est très cher et nécessite des audits réguliers, ce qui n'est pas adapté à une jeune entreprise. MySQL est également une bonne option, mais il est moins sécurisé que PostgreSQL et nécessite InnoDB pour les transactions qui est un prérequis pour la programmation ACID qui est très importante pour la gestion de la concurrence.

3.2 Serveur web

Dans cette section, nous allons vous présenter les solutions disponibles pour la gestion de pages web ainsi que notre recommandation.

3.2.1 Protocoles et normes associés aux serveurs web

Les serveurs web reposent sur plusieurs protocoles standardisés pour la communication avec les navigateurs, le transfert de contenu et la sécurisation des échanges.

Protocole/Norme	Description
HTTP (RFC 7230-7235)	<i>HyperText Transfer Protocol</i> : protocole principal utilisé pour le transfert de pages web. HTTP/1.1 est le standard le plus répandu, HTTP/2 améliore la performance avec le multiplexage.
HTTPS (RFC 2818)	Version sécurisée de HTTP fonctionnant sur TLS. Elle assure le chiffrement des données échangées, ainsi que l'authentification du serveur.
TLS (RFC 5246/8446)	<i>Transport Layer Security</i> : protocole cryptographique assurant la confidentialité et l'intégrité des communications sur Internet. Utilisé par HTTPS, IMAPS, SMTPS, etc.
CGI (RFC 3875)	<i>Common Gateway Interface</i> : norme permettant aux serveurs web d'exécuter des scripts externes pour générer dynamiquement du contenu (très utilisé historiquement avec Apache).

TABLE 5 – Protocoles et normes couramment utilisés avec les serveurs web

3.2.2 Apache

Propriété	Valeur
Éditeur	Apache Software Foundation
Brève description technique	Serveur très polyvalent. De nombreux modules pour différents langages (php, python, perl, ruby)
Licence	Licence Apache (open-source)
OS pris en charge	Windows, Linux, MacOS
Protocoles/normes associées	HTTP, HTTPS, FTP
Qualité documentation	Robuste et exhaustive, mais nécessite une bonne base en administration système
Popularité/Taux d'utilisation	60% des domaines français en 2017 l'utilisent
Avis de la communauté	Fiable et hautement configurable, Apache est simple à déployer, compatible avec tous les environnements, et soutenu par un écosystème riche et éprouvé.

TABLE 6 – Caractéristiques principales de Apache

3.2.3 Nginx

Propriété	Valeur
Éditeur	F5 Inc. (anciennement NGINX, Inc.)
Brève description technique	Serveur web léger et performant, conçu pour gérer de fortes charges, excellent en reverse proxy
Licence	Licence BSD-like (open-source)
OS pris en charge	Linux, BSD, Windows (fonctionnalité partielle), MacOS
Protocoles/normes associées	HTTP, HTTPS, SMTP, POP3, IMAP
Qualité documentation	Complète et bien structurée, avec de nombreux exemples mais parfois trop concise pour les débutants
Popularité/Taux d'utilisation	Environ 33% des sites web mondiaux, utilisé par des géants comme Netflix, Dropbox, WordPress
Avis de la communauté	Léger, extrêmement rapide, et facile à configurer pour les besoins modernes, Nginx est très apprécié pour ses performances et sa fiabilité en production.

TABLE 7 – Caractéristiques principales de Nginx

3.2.4 Étude comparative

Ces deux serveurs HTTP peuvent être pertinents dans le cadre de notre projet.

Cependant, nous avons décidé de choisir **Nginx** pour les raisons suivantes :

- Il est open-source, léger et extrêmement performant, ce qui en fait une solution idéale pour un environnement avec plusieurs utilisateurs simultanés.
- Il gère très bien les fichiers statiques et propose des fonctionnalités natives de reverse proxy et de load balancing.
- Il est très populaire et utilisé dans des environnements professionnels à grande échelle (Netflix, Dropbox), ce qui garantit une documentation abondante et des retours d'expérience solides.
- Il est facile à configurer pour des usages modernes tout en étant peu gourmand en ressources système.

Tandis qu'**Apache HTTP Server**, bien que très fiable et modulable, est plus complexe à configurer, et légèrement moins performant pour la gestion de fichiers statiques. Sa structure plus ancienne peut également rendre son déploiement moins agile dans un environnement web moderne. Cela le rend un peu moins adapté pour une jeune entreprise souhaitant simplicité et efficacité immédiate dans la mise en place de services web.

3.3 Service de messagerie

Dans cette section, nous allons vous présenter les solutions disponibles pour la gestion d'un service de messagerie ainsi que notre recommandation.

3.3.1 Protocoles et normes associés aux services de messagerie

Les services de messagerie s'appuient sur plusieurs protocoles normalisés, chacun jouant un rôle spécifique dans l'envoi, la réception ou la consultation des courriels.

Protocole (RFC)	Description
SMTP (RFC 5321)	<i>Simple Mail Transfer Protocol</i> : protocole utilisé pour l'envoi de courriels depuis un client vers un serveur ou entre serveurs de messagerie. Il ne permet pas la consultation des courriels.
POP (RFC 1939)	<i>Post Office Protocol version 3</i> : protocole permettant à un client de récupérer les courriels depuis un serveur. Une fois les messages téléchargés, ils sont en général supprimés du serveur. Idéal pour une consultation locale unique.
IMAP (RFC 3501)	<i>Internet Message Access Protocol</i> : protocole plus moderne que POP, permettant une gestion des courriels directement sur le serveur. Il est conçu pour permettre l'accès simultané depuis plusieurs dispositifs et conserve les messages sur le serveur.

TABLE 8 – Principaux protocoles utilisés dans les services de messagerie

3.3.2 Postfix

Propriété	Valeur
Éditeur	Wietse Venema (IBM Research)
Brève description technique	MTA (Mail Transfer Agent) moderne, sécurisé et performant, conçu comme une alternative à Sendmail
Licence	Licence IBM Public License (open-source)
OS pris en charge	Principalement Unix/Linux
Protocoles/normes associées	SMTP, SMTPS, STARTTLS, Militer
Qualité documentation	Documentation très complète, avec une forte communauté et de nombreuses ressources d'aide
Popularité/Taux d'utilisation	L'un des MTA les plus utilisés sur les serveurs Linux
Avis de la communauté	Réputé pour sa fiabilité, sa sécurité et sa facilité de configuration, Postfix est considéré comme le standard de facto pour les environnements de messagerie modernes.

TABLE 9 – Caractéristiques principales de Postfix

3.3.3 Exim

Propriété	Valeur
Éditeur	University of Cambridge
Brève description technique	MTA hautement configurable, historiquement populaire sur les systèmes Debian
Licence	GNU General Public License (open-source)
OS pris en charge	Principalement Unix/Linux
Protocoles/normes associées	SMTP, TLS, SPF, DKIM, DMARC
Qualité documentation	Complète, mais parfois ardue à exploiter du fait de la complexité de certaines configurations
Popularité/Taux d'utilisation	Préinstallé sur de nombreuses distributions Linux, très présent dans les hébergements mutualisés
Avis de la communauté	Très puissant et flexible, mais jugé complexe à configurer et maintenir. Son architecture modulaire peut être un avantage comme un inconvénient selon les besoins.

TABLE 10 – Caractéristiques principales de Exim

3.3.4 Étude comparative

Les deux solutions présentées sont robustes et adaptées à une utilisation professionnelle.

Cependant, nous avons décidé de recommander **Postfix**, pour les raisons suivantes :

- Il est reconnu pour sa sécurité, sa robustesse et sa simplicité de configuration, ce qui le rend idéal pour une entreprise souhaitant mettre en place rapidement un service fiable.
- Il est largement adopté dans l'industrie, avec une documentation abondante et une communauté active, facilitant le support et les bonnes pratiques.
- Son architecture modulaire permet des extensions via des outils comme SpamAssassin, ClamAV, ou encore Dovecot pour la partie IMAP/POP.
- Il est optimisé pour la performance tout en étant léger, ce qui en fait un bon choix même sur des infrastructures modestes.

Tandis qu'**Exim**, bien que très puissant et flexible, peut rapidement devenir complexe à gérer à mesure que les besoins en sécurité ou en routage évoluent. Cela représente une courbe d'apprentissage plus importante, souvent inutile pour une structure en démarrage ou avec peu de ressources dédiées à l'administration système. Ainsi, Postfix offre un meilleur compromis entre simplicité, performance et sécurité pour un déploiement professionnel rapide et fiable.

4 Conclusion

Dans ce rapport, nous avons présenté de multiples solutions logicielles pour répondre aux besoins de l'entreprise. Même si nous avons émis des recommandations, chacune des solutions présentées peuvent être utilisées dans le cadre de notre projet avec des résultats similaires. Les recommandations essayant principalement de tenir compte du contexte de l'entreprise, si celui-ci est amené à changer, il faudra peut-être reconsidérer les choix effectués.