

# Documentation Technique Complète - Projet Holodeck Starfleet

Infrastructure de Développement Web pour les Ingénieurs de Starfleet

---

## Table des Matières

1. [Vue d'Ensemble du Projet](#)
  2. [Architecture Technique](#)
  3. [Infrastructure Réseau](#)
  4. [Services Implémentés](#)
  5. [Configuration et Sécurité](#)
  6. [Tests et Validation](#)
  7. [Scripts et Automatisation](#)
  8. [Documentation et GitHub](#)
  9. [Résultats et Achievements](#)
  10. [Perspectives et Améliorations](#)
- 

## 1. Vue d'Ensemble du Projet

### 1.1 Contexte et Objectifs

Le projet Holodeck vise à créer une infrastructure de développement web complète pour les ingénieurs de Starfleet, simulant un environnement professionnel avec des services multiples et une architecture réseau complexe.

#### Objectifs principaux :

- Déployer une infrastructure virtualisée complète
- Implémenter des services web avec différentes versions PHP (7.4 et 8.2)
- Configurer des services réseau (DNS/DHCP) pour un domaine personnalisé
- Assurer la sécurité avec HTTPS et authentification LDAP
- Créer une documentation technique professionnelle

### 1.2 Contraintes Techniques

#### Conformément aux directives de Starfleet :

- Pas de compte sudo (utilisation root directe)
- Pare-feu restrictif autorisant uniquement les ports requis

- Serveur web Nginx en HTTPS obligatoire
  - Dernières versions de PHP, MariaDB, et Nginx (hors dépôts Debian)
  - Coexistence PHP 7.x et 8.x
  - Authentification LDAP pour les utilisateurs web
- 

## 2. Architecture Technique

### 2.1 Infrastructure Virtualisée

#### VM Serveur (holodeck-server) :

- OS : Debian 12 sans interface graphique
- RAM : 4GB | CPU : 2 vCPU | Disque : 32GB
- Interfaces réseau : 2 cartes (WAN + LAN)
- Rôle : Serveur principal multi-services

#### VM Client (HolodeckClient) :

- OS : Debian 12 avec interface GNOME
- RAM : 2GB | CPU : 2 vCPU | Disque : 16GB
- Interface réseau : 1 carte LAN uniquement
- Rôle : Client de test et validation

### 2.2 Topologie Réseau

```
graph TD
    Internet[Internet] --> Gateway["[192.168.81.2] Gateway VMware (WAN)"]
    Gateway --> ServerWAN["[192.168.81.134] ens33 (WAN) - holodeck-server"]
    ServerWAN --> ServerLAN["[192.168.100.253] ens37 (LAN) - holodeck-server"]
    ServerLAN -- "DHCP + NAT" --> Clients["[192.168.100.10-100] Clients DHCP"]
    Clients --> Client["[192.168.100.10] HolodeckClient"]
```

#### Résolution des conflits réseau :

- Problème initial : IP 192.168.100.1 en conflit entre gateway et serveur
- Solution : Migration serveur vers 192.168.100.253

- Mise à jour des zones DNS et configuration DHCP
- 

### 3. Infrastructure Réseau

#### 3.1 Service DHCP

Configuration (/etc/dhcp/dhcpd.conf) :

- Domaine : starfleet.lan
- Serveur DHCP : 192.168.100.253
- Plage IP : 192.168.100.10-100
- Bail : 600s (défaut), 7200s (maximum)
- DNS : 192.168.100.253 (local), 192.168.81.2 (externe)

Résolution problème DHCP :

- Diagnostic : Service "Network is down" dû au conflit IP
- Solution : Reconfiguration complète avec IP dédiée serveur
- Validation : Client reçoit automatiquement IP dans la plage

#### 3.2 Service DNS (BIND9)

Configuration maître (/etc/bind/named.conf.local) :

- Zone forward : starfleet.lan
- Zone reverse : 100.168.192.in-addr.arpa
- Serveur autoritaire : 192.168.100.253

Enregistrements DNS principaux :

```
dns
@      IN  A    192.168.100.253
ns     IN  A    192.168.100.253
www7   IN  A    192.168.100.253
www8   IN  A    192.168.100.253
php     IN  A    192.168.100.253
admin  IN  A    192.168.100.253
vscore IN  A    192.168.100.253
dashboard IN A    192.168.100.253
```

#### 3.3 Routage et NAT

Configuration avancée :

- IP Forwarding activé : `net.ipv4.ip_forward=1`
  - Règles iptables NAT pour connectivité Internet
  - Forward LAN vers WAN avec masquerading
  - Politique de sécurité restrictive
- 

## 4. Services Implémentés

### 4.1 Serveur Web Nginx

Caractéristiques :

- Version : Dernière version (non dépôt Debian)
- Configuration HTTPS obligatoire
- Certificats SSL auto-signés pour starfleet.lan
- Virtual hosts multiples pour différents services

Sites web configurés :

- `www7.starfleet.lan` : Environment PHP 7.4 (legacy)
- `www8.starfleet.lan` : Environment PHP 8.2 (moderne)
- `php.starfleet.lan` : phpMyAdmin pour gestion BDD
- `admin.starfleet.lan` : Webmin pour administration système
- `vscode.starfleet.lan` : Visual Studio Code Server
- `dashboard.starfleet.lan` : Dashboard personnalisé Holodeck

### 4.2 PHP Multi-Version

Coexistence PHP 7.4 et 8.2 :

- PHP-FPM 7.4 : Sites legacy et compatibilité
- PHP-FPM 8.2 : Applications modernes
- Configuration séparée par virtual host
- Extensions installées : MySQL, GD, Curl, XML, etc.

### 4.3 Base de Données MariaDB

Configuration :

- Version : MariaDB 12.0.2 (dernière version)
- Port : 3306 (standard)
- Utilisateurs configurés :

- root : Administration complète
- phpmyadmin : Accès via interface web
- Utilisateurs applicatifs spécifiques

## 4.4 Services Additionnels

### Webmin :

- Interface d'administration système via web
- Accès sécurisé HTTPS
- Gestion des services, utilisateurs, configurations

### Visual Studio Code Server :

- Environnement de développement web intégré
- Accès via navigateur sur vscore.starfleet.lan
- Configuration avec authentification

### Dashboard Personnalisé :

- Interface de monitoring Holodeck
  - Authentification utilisateurs LDAP
  - Affichage statut services et métriques
- 

## 5. Configuration et Sécurité

### 5.1 Certificats SSL/TLS

#### Génération certificats auto-signés :

- CA racine pour starfleet.lan
- Certificats wildcard : \*.starfleet.lan
- Configuration Nginx avec SSL optimisé
- Redirection HTTP → HTTPS automatique

### 5.2 Authentification LDAP

#### Annuaire LDAP configuré :

- Base DN : dc=starfleet,dc=lan
- Utilisateurs de test :
  - spock / VulcanLogic123!

- samet / Holodeck2025!
- Intégration web applications

## 5.3 Sécurité Réseau

Pare-feu (iptables) :

- Politique DROP par défaut
- Ports autorisés :
  - 22 (SSH)
  - 53 (DNS)
  - 67 (DHCP)
  - 80/443 (HTTP/HTTPS)
  - 3306 (MariaDB - local)
  - 21 (FTP avec SSL/TLS)

## 5.4 Serveur FTP Sécurisé

Configuration vsftpd :

- SSL/TLS obligatoire
  - Chroot sur répertoire web
  - Utilisateurs dédiés au transfert fichiers
  - Logs sécurisés et monitoring
- 

## 6. Tests et Validation

### 6.1 Tests de Connectivité Réseau

Validation DHCP/DNS :

```
bash

# Client obtient IP automatiquement
ip addr show → 192.168.100.10/24

# Résolution DNS fonctionnelle
nslookup www8.starfleet.lan → 192.168.100.253

# Connectivité Internet via NAT
ping 8.8.8.8 → Successful
```

## 6.2 Tests Services Web

### Validation HTTPS :

- <https://admin.starfleet.lan> → Webmin accessible
- <https://php.starfleet.lan> → phpMyAdmin opérationnel
- <https://www7.starfleet.lan> → Site PHP 7.4 fonctionnel
- <https://www8.starfleet.lan> → Site PHP 8.2 fonctionnel
- <https://dashboard.starfleet.lan> → Dashboard personnalisé

### Validation bases données :

- Connexion phpMyAdmin réussie
- Requêtes SQL opérationnelles
- Utilisateurs et permissions corrects

## 6.3 Tests Performance et Sécurité

### Métriques de performance :

- Temps de réponse DNS < 10ms
- Réponse DHCP < 50ms
- Chargement pages web < 200ms
- SSL handshake optimisé

### Validation sécurité :

- Certificats SSL valides
  - Authentification LDAP opérationnelle
  - Pare-feu restrictif fonctionnel
  - Logs de sécurité configurés
- 

## 7. Scripts et Automatisation

### 7.1 Scripts de Maintenance

#### holodeck-backup.sh :

- Sauvegarde configurations système
- Export bases de données
- Rotation automatique des sauvegardes

- Vérification intégrité

**verification-services.sh :**

- Monitoring automatique des services
- Vérification statut : nginx, php-fpm, mariadb, bind9, dhcp
- Rapports colorés avec statuts visuels
- Alertes en cas de problème

## 7.2 Scripts de Déploiement

**installation-complete.sh :**

- Installation automatisée des services
- Configuration réseau automatique
- Déploiement certificats SSL
- Tests post-installation

**post-import-config.sh :**

- Reconfiguration après import VM
- Adaptation adresses IP
- Redémarrage services nécessaires
- Validation fonctionnement

## 7.3 Monitoring et Logs

**Surveillance continue :**

- Logs centralisés dans /var/log/
- Monitoring ressources système
- Alertes automatiques par email
- Dashboard temps réel

---

## 8. Documentation et GitHub

### 8.1 Repository GitHub

**Structure organisée :**



holodeck/

├── README.md (documentation principale)

├── docs/ (guides détaillés)

| ├── guide-installation.md

| ├── guide-utilisateur.md

| ├── depannage.md

| └── export-vm.md

├── scripts/ (automatisation)

| ├── verification-services.sh

| ├── holodeck-backup.sh

| └── installation-complete.sh

├── configs/ (configurations de référence)

├── screenshots/ (captures d'écran services)

| └── services/

└── LICENSE

## 8.2 Documentation Technique

**Guides complets :**

- **Installation** : Procédure pas-à-pas déploiement
- **Utilisation** : Guide utilisateur final
- **Dépannage** : Solutions problèmes courants
- **Export VM** : Procédures sauvegarde et migration

**Captures d'écran services :**

- Interface Webmin opérationnelle
- phpMyAdmin avec base données
- Sites PHP 7.4 et 8.2 fonctionnels
- Dashboard personnalisé Holodeck
- VS Code Server intégré








## 8.3 Versioning et Collaboration







**Gestion Git professionnelle :**

- Commits conventionnels (feat:, fix:, docs:)
  - Branches structurées
  - README avec badges et métriques
  - Documentation complète en français
-

## 9. Résultats et Achievements

### 9.1 Objectifs Atteints

**Infrastructure complète :**  2 VM Debian configurées et opérationnelles  Services réseau DHCP/DNS pour domaine starfleet.lan  Serveur web Nginx avec HTTPS et multi-sites   
Coexistence PHP 7.4 et 8.2 fonctionnelle  Base données MariaDB avec interface phpMyAdmin   
Authentification LDAP implémentée  Certificats SSL auto-signés déployés

**Services additionnels :**  Webmin pour administration système  Visual Studio Code Server intégré  Dashboard personnalisé Holodeck  Serveur FTP sécurisé SSL/TLS  Scripts automatisés et monitoring  Documentation technique complète

### 9.2 Compétences Développées

**Administration système Linux :**

- Configuration services réseau avancés
- Gestion multi-version PHP avec Nginx
- Sécurisation SSL/TLS et certificats
- Debugging et résolution problèmes réseau

**Développement et DevOps :**

- Scripts Bash automatisés
- Versioning Git professionnel
- Documentation technique structurée
- Monitoring et maintenance automatisés

**Sécurité informatique :**

- Pare-feu iptables configuré
- Authentification LDAP centralisée
- Chiffrement SSL/TLS bout-en-bout
- Audit sécurité et logs

### 9.3 Défis Relevés

**Problèmes résolus :**

1. **Conflit IP DHCP/Gateway :** Migration architecture réseau
2. **Authentification GitHub :** Configuration tokens sécurisés
3. **Coexistence PHP multi-version :** Configuration FPM séparée

4. **Résolution DNS locale** : Zones BIND9 personnalisées

5. **Intégration services** : Virtual hosts Nginx optimisés

---

## 10. Perspectives et Améliorations

### 10.1 Évolutions Techniques Possibles

#### Containerisation :

- Migration vers Docker/Podman
- Orchestration avec Docker Compose
- Images personnalisées optimisées

#### Monitoring avancé :

- Intégration Prometheus/Grafana
- Dashboard métriques temps réel
- Alertes automatisées

#### Haute disponibilité :

- Réplication base données
- Load balancing Nginx
- Clustering services critiques

### 10.2 Extensions Fonctionnelles

#### Sécurité renforcée :

- Certificats Let's Encrypt automatiques
- Authentification deux facteurs
- Audit sécurité automatisé

#### Développement :

- CI/CD avec GitLab/Jenkins
- Environnements de développement isolés
- Tests automatisés

### 10.3 Optimisations Performance

#### Infrastructure :

- Optimisation configurations PHP-FPM

- Cache Redis/Memcached
- CDN pour ressources statiques

#### **Monitoring :**

- APM (Application Performance Monitoring)
  - Logging centralisé ELK Stack
  - Métriques business personnalisées
- 

## **Conclusion**

Le projet Holodeck représente une implémentation complète et professionnelle d'une infrastructure de développement web moderne. Avec ses 8 services configurés, son architecture réseau complexe, sa documentation technique exhaustive et ses scripts d'automatisation, il démontre une maîtrise approfondie de l'administration système Linux et des technologies web.

L'intégration GitHub avec documentation structurée, captures d'écran et scripts de déploiement en fait un projet de référence pour la formation d'ingénieurs système. Les défis techniques relevés (résolution conflits réseau, coexistence PHP multi-version, sécurisation SSL/TLS) illustrent une approche méthodique de résolution de problèmes complexes.

Ce projet constitue une base solide pour des évolutions futures vers des architectures microservices, du monitoring avancé, ou des déploiements cloud, démontrant la scalabilité de l'architecture mise en place.

---

**Projet développé pour la formation Starfleet Engineering**

**Repository GitHub :** <https://github.com/samet-ari/holodeck>

**Technologies :** Debian 12, Nginx, PHP 7.4/8.2, MariaDB, BIND9, DHCP, SSL/TLS, LDAP