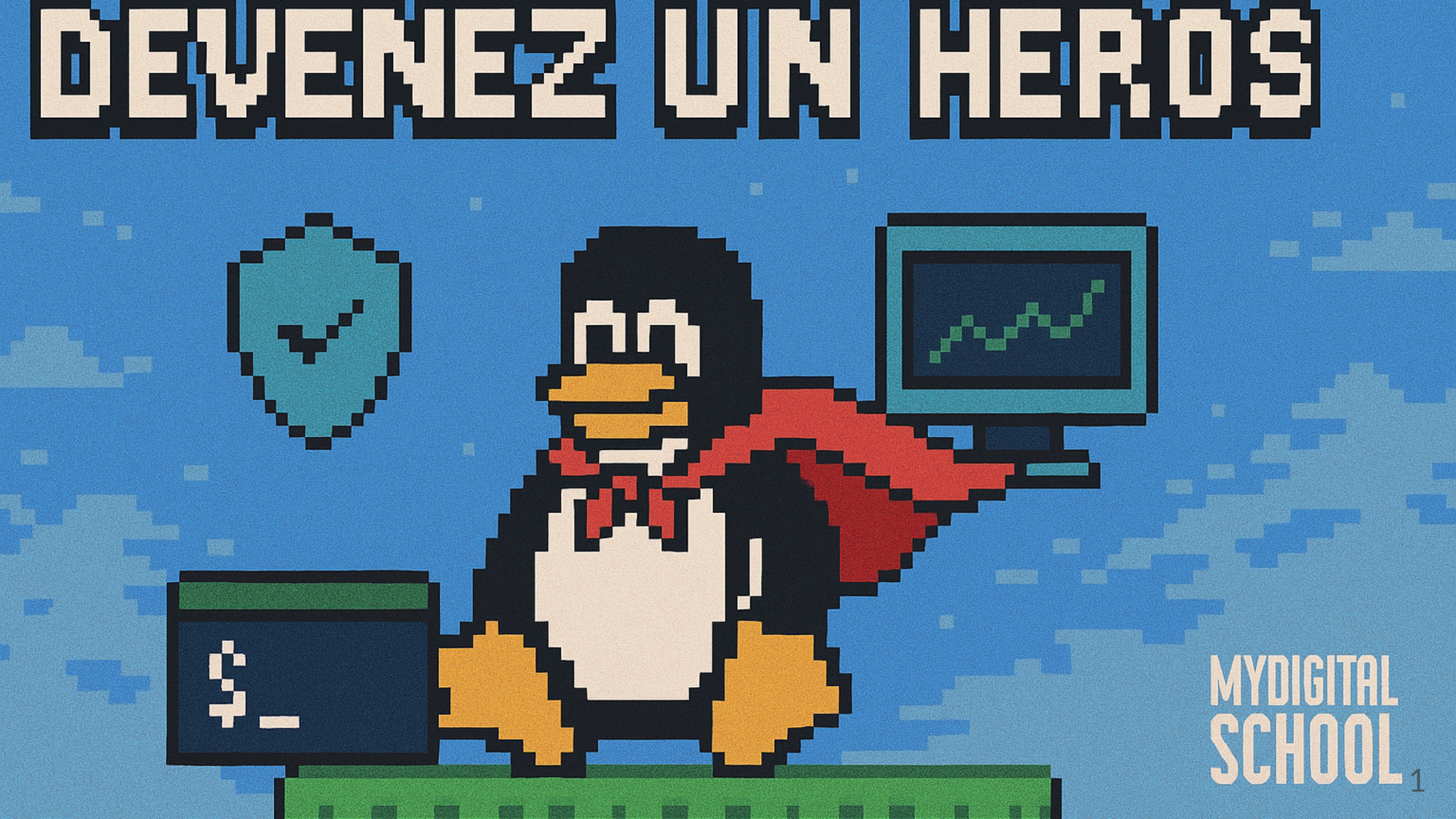
DEVENEZ UN HEROS



MYDIGITAL SCHOOL

🐧 Projet Linux Avancé

"Mon Premier Serveur Sécurisé & Monitoré"

🧠 1 Journée – Travail individuel – Évaluation notée

🎯 Objectif pédagogique

Mettre en œuvre **les compétences clés du module GNU/Linux avancé** :

- Déploiement d'un serveur Linux
- Sécurisation (pare-feu, SSH, fail2ban)
- Monitoring (Netdata, Grafana…)
- Mise en place d’un reverse proxy (Nginx + Docker)
- Tests de charge
- Maintenance (scripts, crontab, sauvegarde)

📜 Énoncé global

Vous êtes administrateur système junior et votre mission est de préparer un serveur Linux pour héberger une future application web.

Vous devez **concevoir, sécuriser, superviser et tester** ce serveur en suivant des pratiques professionnelles.

Travail individuel sur un VPS attribué par l'école.

📦 Étape 0 – Outils de base à installer

Installez les outils suivants et documentez-les dans un fichier `README.md` :

- `htop`, `curl`, `wget`, `zsh`, `oh-my-zsh`
- `git`, `nano`, `vim`, `tree`, `lsof`, `ncdu`
- `ufw`, `fail2ban`, `openssh-server`, `sudo`, `cron`
- `rsync`, `net-tools`, `iproute2`, `screen` ou `tmux`, `lsb-release`

📦 Étape 0 – Outils de base à installer

✍️ **Consigne :** Pour chaque outil, indiquez dans le `README.md` :

1. Sa commande d’installation
2. Son rôle
3. Une commande d’utilisation typique

🧩 Étape 1 – Déploiement initial

- Mettre à jour le système
- Créer un utilisateur admin avec `sudo`
- Activer SSH avec **authentification par clé uniquement**
- Interdire la connexion root via SSH

🛡️ Étape 2 – Sécurisation

- Activer et configurer le pare-feu `ufw`
- Installer et configurer `fail2ban` pour SSH
- Interdire les ports inutiles, autoriser 22/80/443 uniquement

📊 Étape 3 – Monitoring

- Installer `Netdata` **ou** `Prometheus + Grafana`
- Superviser CPU, RAM, disque, réseau
- Bonus : configurer une alerte espace disque > 80 %

🌐 Étape 4 – Serveur Web & Reverse Proxy

- Installer et configurer `Nginx`
- Créer un reverse proxy vers un conteneur Docker (`nginx` ou `node`)
- Générer un certificat HTTPS avec Certbot (Let’s Encrypt)

🧪 Étape 5 – Test de charge

- Installer `ab` (Apache Benchmark) ou `wrk`
- Lancer un test de charge : 1000 requêtes, 10 utilisateurs
- Documenter les résultats (temps de réponse, débit)

💾 Étape 6 – Maintenance & Sauvegarde

- 🔹 Écrire un script Bash qui :

- Sauvegarde `/etc`, `/var/log`, `/home/username`
- Compresse le tout dans `/home/backup/backup_DATE.tar.gz`
  - 🔹 Automatiser l'exécution toutes les 6h avec `cron`

📂 Étape 7 – Documentation

Dans votre répertoire personnel sur le VPS :

- Créez un fichier `README.md`
- Documentez chaque étape (commandes, captures, explication)
- Versionnez le projet (Documentation) avec Git

📤 Possibilité d'héberger sur GitHub ou GitLab personnel

🎤 Restitution finale

Chaque étudiant doit :

- Donner accès SSH à son VPS
- Présenter oralement son travail (5 min max)
- Expliquer la logique de son script de sauvegarde
- Montrer le monitoring et le reverse proxy

📊 Barème d’évaluation

Critère	Points
Installation des outils de base + doc	15 %
Sécurisation du VPS (SSH, UFW, fail2ban)	20 %
Mise en place du monitoring	15 %
Reverse proxy + Docker + HTTPS	20 %
Script de sauvegarde + cron	15 %
Qualité de la documentation et présentation	15 %

📚 Ressources utiles

- 🔗 doc.ubuntu-fr.org
- 🔗 netdata.cloud
- 🔗 certbot.eff.org
- 🔗 grafana.com
- 🔗 nmap.org

✅ Recommandations

- 🧪 Testez chaque étape dès qu’elle est terminée
- 📸 Prenez des captures d’écran (Netdata, Grafana, etc.)
- 💾 Faites une sauvegarde finale de votre VPS à la fin
- 📌 Soignez votre `README.md`, il sera évalué

🎯 Bon courage sysadmins en herbe !