perplexity

Ajouter les règles habituelles dans un environement tel que décrit avant

Voici les **règles de firewall OPNsense recommandées** pour un environnement d'entreprise moderne, découpé en zones (LAN, DMZ, IoT, etc.), avec microservices, cloud, accès distants, et intégration DevOps. Ces règles sont à placer dans le dossier dédié, par exemple : `infrastructure/network/opnsense/terraform/firewall_rules.tf`

1. Principes généraux (État de l'art)

- **Politique par défaut restrictive** : tout est bloqué sauf ce qui est explicitement autorisé [1] [2].
- **Segmentation stricte** : séparation des réseaux (LAN, DMZ, IoT, VLANs, etc.) [1] [2].
- **Administration sécurisée** : accès à l'admin OPNsense limité à des IP/segments de confiance, jamais exposé sur le WAN [1].
- **Logs et alertes** : journalisation de tous les accès refusés et des flux critiques [1].
- **Règles explicites et documentées** : chaque règle a une description claire [1] [3].

2. Règles types à intégrer

A. WAN (Internet vers interne)

- **Bloquer tout par défaut** (action : Block/Reject, source : any, destination : any).
- **Autoriser uniquement les services exposés (ex : HTTPS sur DMZ, VPN)** :
  - Pass, Protocol: TCP, Source: any, Destination: [IP publique DMZ], Port: 443 (HTTPS)
  - Pass, Protocol: UDP, Source: any, Destination: [IP VPN], Port: [1194/500/4500] (VPN)

B. LAN (interne vers Internet)

- **Autoriser HTTP/HTTPS sortant** :
  - Pass, Protocol: TCP, Source: LAN net, Destination: any, Port: 80, 443, Description: "Allow web browsing"
- **Autoriser DNS vers firewall** :
  - Pass, Protocol: TCP/UDP, Source: LAN net, Destination: [LAN address], Port: 53, Description: "Allow DNS to firewall"
- **Bloquer tout le reste** (Block all, Source: LAN net, Destination: any)

C. LAN (interne vers DMZ, IoT, autres VLANs)

- **Bloquer par défaut le trafic inter-VLAN** :
  - Block, Protocol: any, Source: LAN net, Destination: [Private_IP_Ranges alias sauf LAN], Description: "Block inter-VLAN traffic" [1]
- **Autoriser les flux nécessaires (ex : accès API DMZ)** :
  - Pass, Protocol: TCP, Source: LAN net, Destination: [DMZ API], Port: 8080, Description: "Allow access to DMZ API"

D. DMZ

- **Autoriser uniquement les flux nécessaires depuis Internet et LAN** (ex : HTTPS, SSH restreint, monitoring).
- **Bloquer tout le reste**.

E. IoT

- **Bloquer tout accès entrant depuis Internet**.
- **Autoriser uniquement les flux nécessaires vers les microservices ou brokers IoT**.
- **Bloquer l'accès des IoT vers LAN/DMZ sauf exceptions explicites**.

F. Administration OPNsense

- **Autoriser uniquement les IPs d'administration** (bastion, VPN, subnet admin) vers l'interface d'admin (HTTPS/SSH) [1].
- **Bloquer l'accès à l'admin depuis le WAN**.

G. NAT

- **NAT sortant automatique** (par défaut) [2].
- **Règles de port forwarding uniquement pour les services explicitement exposés**.

H. QoS et gestion de la bande passante

- **Prioriser les flux critiques (VoIP, monitoring, etc.)** si besoin [3].

3. Exemples de règles (Terraform style)

```
# Bloquer tout par défaut sur WAN
resource "opnsense_firewall_filter" "wan_block_all" {
  interface   = "WAN"
  action      = "block"
  source      = "any"
  destination = "any"
  description = "Block all inbound WAN traffic"
}
```

```
# Autoriser HTTPS sur DMZ
resource "opnsense_firewall_filter" "wan_allow_https_dmz" {
  interface   = "WAN"
  action      = "pass"
  protocol    = "tcp"
  destination = "DMZ_IP"
  destination_port = "443"
  description = "Allow HTTPS to DMZ"
}

# Autoriser HTTP/HTTPS sortant depuis LAN
resource "opnsense_firewall_filter" "lan_allow_web" {
  interface   = "LAN"
  action      = "pass"
  protocol    = "tcp"
  source      = "LAN_net"
  destination = "any"
  destination_port = ["80", "443"]
  description = "Allow HTTP/HTTPS out"
}

# Autoriser DNS vers firewall
resource "opnsense_firewall_filter" "lan_allow_dns" {
  interface   = "LAN"
  action      = "pass"
  protocol    = ["tcp", "udp"]
  source      = "LAN_net"
  destination = "LAN_address"
  destination_port = "53"
  description = "Allow DNS to firewall"
}

# Bloquer tout le reste sur LAN
resource "opnsense_firewall_filter" "lan_block_all" {
  interface   = "LAN"
  action      = "block"
  source      = "LAN_net"
  destination = "any"
  description = "Block all other LAN traffic"
}

# Bloquer inter-VLAN
resource "opnsense_firewall_filter" "lan_block_intervlan" {
  interface   = "LAN"
  action      = "block"
  source      = "LAN_net"
  destination = "Private_IP_Ranges"
  description = "Block inter-VLAN traffic"
}

# Autoriser admin à l'interface OPNsense
resource "opnsense_firewall_filter" "lan_allow_admin" {
  interface   = "LAN"
  action      = "pass"
  protocol    = "tcp"
  source      = "Admin_IPs"
```

```
  destination = "LAN_address"
  destination_port = ["443", "22"]
  description = "Allow admin access to OPNsense"
}
```

4. Conseils de gestion

- **Documente chaque règle** (description, justification, date de création).
- **Active la journalisation sur les règles critiques**.
- **Teste chaque flux (scans, essais de connexion)** après déploiement.
- **Sauvegarde la configuration** régulièrement.

À placer dans :
`infrastructure/network/opnsense/terraform/firewall_rules.tf`
et à adapter selon la topologie réelle, les IPs, les ports/services effectivement utilisés dans ton environnement microservices, cloud, IoT, etc.[1] [2] [3].

Ces règles couvrent l'essentiel pour un environnement professionnel moderne, sécurisé et segmenté, conforme à l'état de l'art.

⁂

1. https://www.zenarmor.com/docs/network-security-tutorials/how-to-configure-opnsense-firewall-rules
2. https://www.it-connect.fr/tuto-installer-et-configurer-opnsense/
3. https://fr.scribd.com/document/818448768/OPNSENSE-FIREWALL