# MITRE ATT&CK matrix y reglas out-of-the-box para syscalls

# Recomendaciones MITRE ATT&ACK

# MITRE ATT&ACK enterprise matrix

**Mitre Corporation:** organismo sin ánimo de lucro financiado por el gobierno federal de EEUU para dar soporte a varias de sus agencias gubernamentales.

MITRE ATT&ACK
https://attack.mitre.org/

- Tactics
- Techniques / Subtechniques
- Matrices

## Falco MITRE Rule Matrix

| Execution | Persistence | Privilege Escalation | Defense Evasion | Credential Access | Discovery | Lateral Movement | Exfiltration |
|---|---|---|---|---|---|---|---|
| DB program spawned process | Modify Shell Configuration File | Launch Privileged Container | Clear Log Activities | Read sensitive file trusted after startup | Read Shell Configuration File | Launch Privileged Container | System procs network activity |
| Run shell untrusted | Schedule Cron Jobs | Non sudo setuid | Delete Bash History | Read sensitive file untrusted | Read ssh information | Launch Sensitive Mount Container | Interpreted procs inbound network |
| Terminal shell in container | Update Package Repository | | | Search Private Keys or Passwords | Read sensitive file untrusted | Launch Disallowed Container | Interpreted procs outbound network |
| Netcat Remote Code Execution in Container | Write below binary dir<br>Write below monitored dir | | | | Contact K8S API Server From Container | | Unexpected UDP Traffic |
| | Write below etc<br>Write below root<br>Write below rpm database | | | | Launch Suspicious Network Tool in Container | | Launch Suspicious Network Tool in Container |
| | Modify binary dirs<br>Mkdir binary dirs | | | | Launch Suspicious Network Tool on Host | | Launch Suspicious Network Tool on Host |
| | User mgmt binaries | | | | | | |
| | Create files below dev | | | | | | |
| | Launch Package Management Process in Container | | | | | | |
| | Remove Bulk Data from Disk Set | | | | | | |
| | Create Hidden Files or Directories | | | | | | |
| | Setuid or Setgid bit | | | | | | |

https://sysdig.com/blog/mitre-attck-framework-for-container-runtime-security-with-sysdig-falco/

# Otros estándares de seguridad:
# NIST, PCI, CIS

# PCI

Payment Card Industry (**PCI**)

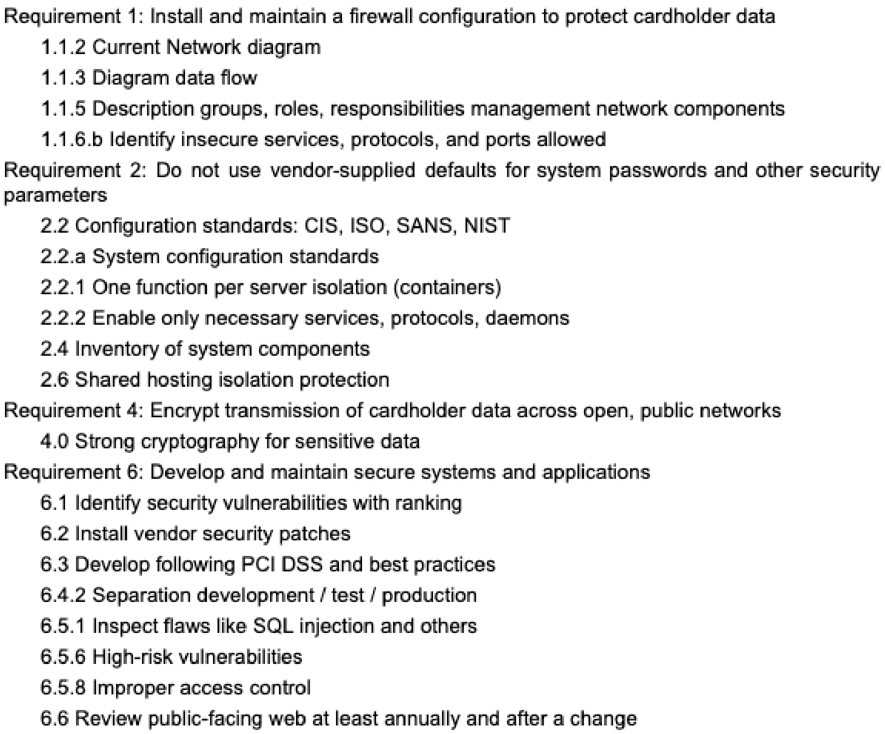
PCI Security Standards Council (**PCI SSC**)
pcisecuritystandards.org

PCI Data Security Standard (**PCI DSS**)

https://es.pcisecuritystandards.org/index.php

# PCI

sysdig.com/resources/whitepapers/a-guide-to-pci-compliance-in-containers-and-kubernetes



Requirement 1: Install and maintain a firewall configuration to protect cardholder data

- 1.1.2 Current Network diagram
- 1.1.3 Diagram data flow
- 1.1.5 Description groups, roles, responsibilities management network components
- 1.1.6.b Identify insecure services, protocols, and ports allowed

Requirement 2: Do not use vendor-supplied defaults for system passwords and other security parameters

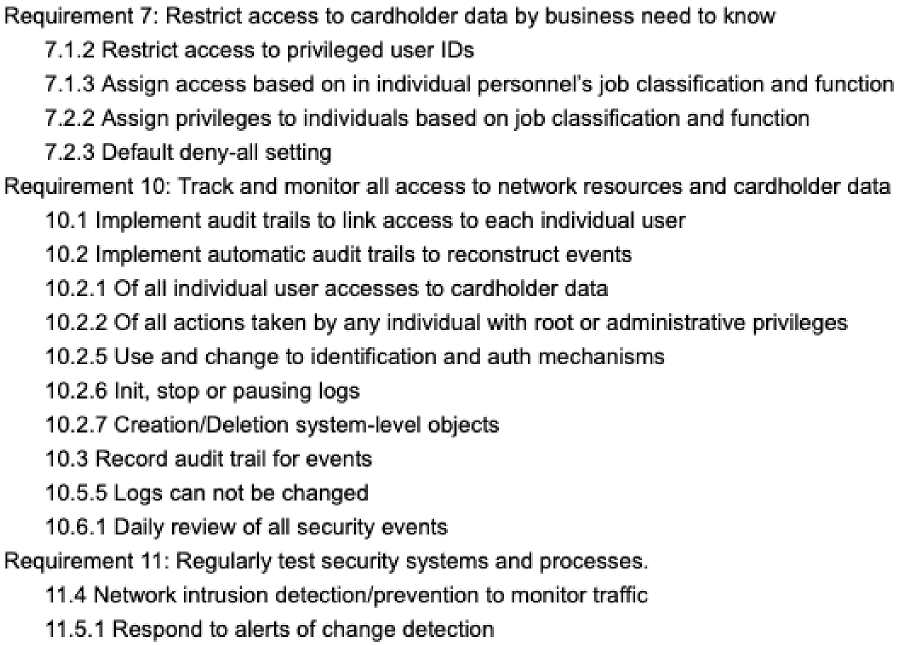
- 2.2 Configuration standards: CIS, ISO, SANS, NIST
- 2.2.a System configuration standards
- 2.2.1 One function per server isolation (containers)
- 2.2.2 Enable only necessary services, protocols, daemons
- 2.4 Inventory of system components
- 2.6 Shared hosting isolation protection

Requirement 4: Encrypt transmission of cardholder data across open, public networks

- 4.0 Strong cryptography for sensitive data

Requirement 6: Develop and maintain secure systems and applications

- 6.1 Identify security vulnerabilities with ranking
- 6.2 Install vendor security patches
- 6.3 Develop following PCI DSS and best practices
- 6.4.2 Separation development / test / production
- 6.5.1 Inspect flaws like SQL injection and others
- 6.5.6 High-risk vulnerabilities
- 6.5.8 Improper access control
- 6.6 Review public-facing web at least annually and after a change



Requirement 7: Restrict access to cardholder data by business need to know

- 7.1.2 Restrict access to privileged user IDs
- 7.1.3 Assign access based on in individual personnel's job classification and function
- 7.2.2 Assign privileges to individuals based on job classification and function
- 7.2.3 Default deny-all setting

Requirement 10: Track and monitor all access to network resources and cardholder data

- 10.1 Implement audit trails to link access to each individual user
- 10.2 Implement automatic audit trails to reconstruct events
- 10.2.1 Of all individual user accesses to cardholder data
- 10.2.2 Of all actions taken by any individual with root or administrative privileges
- 10.2.5 Use and change to identification and auth mechanisms
- 10.2.6 Init, stop or pausing logs
- 10.2.7 Creation/Deletion system-level objects
- 10.3 Record audit trail for events
- 10.5.5 Logs can not be changed
- 10.6.1 Daily review of all security events

Requirement 11: Regularly test security systems and processes.

- 11.4 Network intrusion detection/prevention to monitor traffic
- 11.5.1 Respond to alerts of change detection

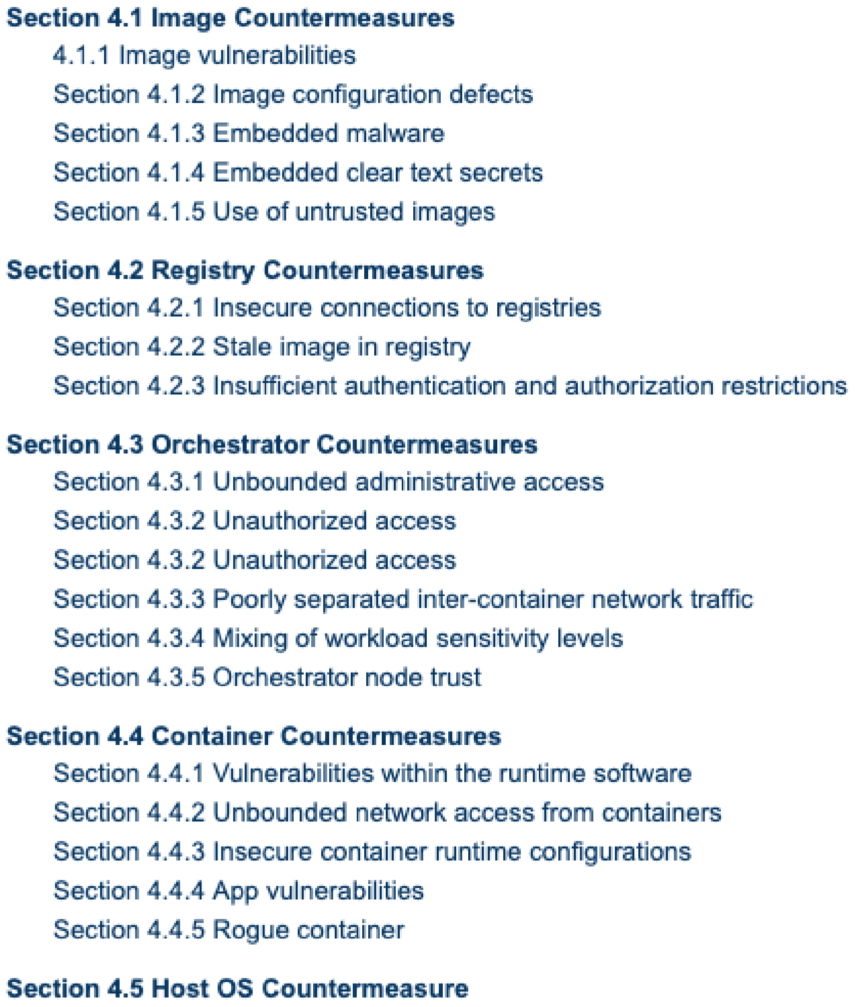
# NIST

National Institute of Standards and Technology (NIST)

**NIST 800-190**
https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-190.pdf

**NIST 800-53**
https://nvd.nist.gov/800-53
Más de 500 controles de seguridad

# NIST 800-190



**Section 4.1 Image Countermeasures**

- 4.1.1 Image vulnerabilities
- Section 4.1.2 Image configuration defects
- Section 4.1.3 Embedded malware
- Section 4.1.4 Embedded clear text secrets
- Section 4.1.5 Use of untrusted images

**Section 4.2 Registry Countermeasures**

- Section 4.2.1 Insecure connections to registries
- Section 4.2.2 Stale image in registry
- Section 4.2.3 Insufficient authentication and authorization restrictions

**Section 4.3 Orchestrator Countermeasures**

- Section 4.3.1 Unbounded administrative access
- Section 4.3.2 Unauthorized access
- Section 4.3.2 Unauthorized access
- Section 4.3.3 Poorly separated inter-container network traffic
- Section 4.3.4 Mixing of workload sensitivity levels
- Section 4.3.5 Orchestrator node trust

**Section 4.4 Container Countermeasures**

- Section 4.4.1 Vulnerabilities within the runtime software
- Section 4.4.2 Unbounded network access from containers
- Section 4.4.3 Insecure container runtime configurations
- Section 4.4.4 App vulnerabilities
- Section 4.4.5 Rogue container

**Section 4.5 Host OS Countermeasure**

CIS Center for Internet Security®

www.cisecurity.org

- Linux Benchmark
- Docker Benchmark
- Kubernetes Benchmark

https://www.cisecurity.org/cis-benchmarks/

# Otros estándares de seguridad

**System and Organization Controls (SOC)**

- AICPA, aicpa.org/interestareas/frc/assuranceadvisoryservices/aicpasoc2report.html
- Designed for service providers storing customer data in the cloud
- Customized to each company

**Health Insurance Portability and Accountability Act (HIPAA)**

- hhs.gov/hipaa/for-professionals/security/laws-regulations/index.html

**(UE) General Data Protection Regulation (GDPR)**

- gdpr-info.eu

# Aportando reglas al proyecto Flaco

# Para enviar una nueva regla a Falco

1. Crear un ticket en el repositorio de Falco para discutir la idea de la necesidad de la regla
2. Configurar firma gpg con git y github https://docs.github.com/en/free-pro-team@latest/github/authenticating-to-github/signing-commits
3. Crear un fork del repositorio https://github.com/falcosecurity/falco
4. Añadir la regla nueva en `falco_rules.yaml`
5. Hacer commit con las opciones `-s -S`
6. Enviar Pull Request hacia el repo principal, indicando el ticket
7. Comentarios de Pull Request: aceptar condiciones de uso de la Cloud Native Computing Foundation