OWASP®

Top 10
Non-Human Identity Risk

glownexus

- Machine to Machine communication forms the core of enterprise software ecosystem.
- It involves the use of API keys, service accounts, certificates or simple credentials being used in SaaS services, Cloud infrastructure or your standard legacy web applications.
- OWASP Top 10 for NHIs is key to understanding the risks and applying appropriate controls in your environments.

glownexus

NHI1:2025 — Improper Offboarding

Risks

- Stale service accounts retain access after decommission
- Orphaned API keys create a persistent attack surface
- Deprecated NHIs go unmonitored and unpatched

Mitigations

- Automate NHI decommissioning tied to service/owner lifecycle
- Run regular audits to detect unused credentials (>90 days)
- Integrate offboarding checks into CI/CD pipelines

NHI2:2025 — Secret Leakage

Risks

- API keys hardcoded in source code or config files
- Tokens exposed in logs, CI output, or public repos
- Leaked secrets enable immediate unauthorized access

Mitigations

- Scan repos with secret detection tools (pre-commit hooks)
- Store secrets in dedicated vaults (e.g., HashiCorp Vault)
- Immediately rotate any exposed credential on discovery

NHI3:2025 — Vulnerable Third-Party NHI

Risks

- Third-party integrations(Integrated Development Environments, Extensions, App Service providers) inherit broad cloud permissions
- Vulnerable vendor NHIs become supply-chain attack vectors
- Overprivileged OAuth apps can access sensitive organisational data

Mitigations

- Take an inventory and review all third-party NHI integrations regularly
- Apply least-privilege scope to every vendor OAuth connection
- Monitor third-party NHI activity for anomalous behavior and have a crisis response plan

glownexus

NHI4:2025 — Insecure Authentication

Risks

- Deprecated authentication methods (basic authentication, static keys) still in use, e.g., OAUTH 1.0
- Long-lived shared passwords with no rotation policy
- Weak credentials vulnerable to brute-force or replay attacks

Mitigations

- Replace static credentials with short-lived tokens (OIDC, JWT)
- Use platform-native identities (AWS IAM Roles, Azure Managed IDs)
- Enforce mutual TLS or SPIFFE/SPIRE for workload authentication

NHI5:2025 — Overprivileged NHI

Risks

- NHIs are assigned far more permissions than their function requires
- Compromised NHI enables full environment takeover
- Broad CI/CD tokens grant attacker access to cloud infrastructure

Mitigations

- Enforce Principle of Least Privilege for every NHI
- Use Just-in-Time (JIT) access grants for elevated tasks
- Conduct periodic access reviews to right-size permissions

NHI6:2025 — Insecure Cloud Deployment Configurations

Risks

- NHIs deployed with default overpermissive IAM roles
- Instance metadata endpoints expose credentials to any process
- Misconfigured workload identities grant cross-account access

Mitigations

- Use IaC security scanners (Checkov, tfsec) to catch misconfigs
- Block IMDS v1; enforce IMDSv2 with token-based access
- Apply Workload Identity Federation instead of static keys

NHI7:2025 — Long-Lived Secrets

Risks

- Credentials older than 90 days greatly expand breach window
- No rotation means a single leak gives persistent access
- Long-lived tokens accumulate in logs and backup stores

Mitigations

- Automate secret rotation with short TTLs (hours, not months)
- Use dynamic secrets (Vault dynamic credentials) where possible
- Alert and auto-disable secrets that exceed age thresholds

NHI8:2025 — Environment Isolation

Risks

- Same NHI credentials used across dev, staging, and production
- Compromise in a lower environment propagates to production
- Shared tokens allow lateral movement between environments

Mitigations

- Provision unique NHIs by environment; never share credentials
- Enforce network and IAM boundaries between environments
- Track cross-environment credential usage in dashboards

NHI9:2025 — NHI Reuse

Risks

- Single NHI credential shared across multiple services
- One compromise grants attacker lateral access to all consumers
- Blast radius of a breach expands with every reuse instance

Mitigations

- Assign a unique identity (SPIFFE ID) to each workload/service
- Use service mesh or SPIRE to issue per-workload certificates
- Audit and eliminate all shared NHI usage in your codebase

NHI10:2025 — Human Use of NHI

Risks

- Admins log in with service accounts, bypassing MFA policies
- Shared API tokens obscure accountability and audit trails
- Attackers leverage NHI credentials for undetected persistence

Mitigations

- Use dedicated human identities for all manual/debug tasks
- Deploy context-aware access policies to block human NHI use
- Educate teams and monitor for anomalous interactive NHI logins

OWASP NHI Top 10:2025 — Summary & Call to Action

The Top 10 Non-Human Identity Risks

- Improper Offboarding
- Secret Leakage
- Vulnerable Third-Party NHI
- Insecure Authentication
- Overprivileged NHI
- Insecure Cloud Deployment Configs
- Long-Lived Secrets
- Environment Isolation
- NHI Reuse
- Human Use of NHI

Immediate Actions — Your Security Checklist

✓ **Audit & offboard** all inactive NHI credentials immediately

✓ **Rotate secrets** and enforce short TTLs; eliminate long-lived tokens

✓ **Scan repos** for leaked secrets and exposed environment variables

✓ **Apply least privilege** to every service account and API key

✓ **Eliminate shared credentials** — unique NHI per service

✓ **Harden cloud IAM** roles; disable default service accounts

✓ **Isolate environments** — no shared NHIs across environments (dev/staging/prod)

✓ **Ban human use of NHI** — enforce dedicated human credentials with MFA

glownexus