# NodeZero Platform: Autonomous Pentest Techniques

Horizon3.ai’s **NodeZero** is a fully automated penetration testing platform that emulates attacker TTPs across networks. It begins with extensive reconnaissance – ping sweeps, port scans, DNS enumeration, and service/OS fingerprinting – to map hosts and identify services[[1]](https://horizon3.ai/intelligence/blogs/nodezero-vs-goad-technical-deep-dive/#:~:text=The%20first%20few%20minutes%20were,across%20three%20Active%20Directory%20domains)[[2]](https://horizon3.ai/nodezero/internal-pentesting/#:~:text=During%20the%20discovery%2Fnetwork%20enumeration%20phase,systems%2C%20web%20servers%2C%20and%20applications). NodeZero can optionally include **OSINT** (public records, domain names, social media, etc.) to discover employee names, domains, and other targets[[3]](https://horizon3.ai/glossary-vocab/osint/#:~:text=OSINT%20is%20the%20acronym%20for,just%20as%20an%20attacker%20might)[[4]](https://docs.horizon3.ai/portal/deployment_strategy/#:~:text=6). The platform supports “Intelligent Scope” discovery: starting from one host and organically expanding across subnets as new hosts are found[[5]](https://docs.horizon3.ai/portal/deployment_strategy/#:~:text=Intelligent%20Scope). During discovery it fingerprints routers, domain controllers, VMware servers, databases, IoT devices, cloud resources and more[[2]](https://horizon3.ai/nodezero/internal-pentesting/#:~:text=During%20the%20discovery%2Fnetwork%20enumeration%20phase,systems%2C%20web%20servers%2C%20and%20applications). Scanning parameters (packets-per-second, host discovery rate) are configurable so administrators can slow NodeZero to reduce noise. For internal tests, NodeZero can perform LLMNR/NetBIOS poisoning and NTLM relaying to capture credentials on‐the‐wire[[6]](https://docs.horizon3.ai/knowledge_base/attack_configurations/#:~:text=match%20at%20L539%20Option%20Description,none)[[7]](https://horizon3.ai/nodezero/internal-pentesting/#:~:text=%2A%20Exposed%20credentials%C2%A0include%C2%A0Man,run%20regular%C2%A0password%20audits%20with%20NodeZero).

## Vulnerability Exploitation & Attack Modules

After mapping the network, NodeZero automatically tests for **known vulnerabilities and misconfigurations**. Its built-in modules cover a wide range of CVEs, default-credential checks, and protocol attacks. For example, NodeZero will try exploits like **BlueKeep (CVE-2019-0708)**, EternalBlue/EternalRomance (MS17-010), Cisco Smart Install (CVE-2018-0171), IIS “Exploding Can” (CVE-2017-7269), Heartbleed, HP iLO RCE (CVE-2017-12542), MS08-067, and many others[[8]](https://docs.horizon3.ai/knowledge_base/attack_configurations/#:~:text=Option%20Description%20Risk%20Bluekeep%20%28CVE,high)[[9]](https://docs.horizon3.ai/knowledge_base/attack_configurations/#:~:text=EternalSynergy%2C%20and%20EternalRomance,based%20buffer%20overflow.%20low). It also tests default logins on FTP, MSSQL, MongoDB, MySQL, PostgreSQL, SNMP, SSH, Telnet and HTTP services[[10]](https://docs.horizon3.ai/knowledge_base/attack_configurations/#:~:text=Option%20Description%20Risk%20FTP%20Enables,default%20credentials%20against%20PostgreSQL%20databases). Network attack modules include LLMNR/NetBIOS/NBT-NS poisoning and NTLM relay[[6]](https://docs.horizon3.ai/knowledge_base/attack_configurations/#:~:text=match%20at%20L539%20Option%20Description,none).

NodeZero is especially strong on **Active Directory attacks**. It attempts credential-gathering and exploitation such as Kerberoasting, AS-REP roasting, Kerberos “noPAC” (unconstrained delegation), AD CS abuse (Certifried), and DCSync/Golden Ticket attacks[[11]](https://horizon3.ai/intelligence/blogs/nodezero-vs-goad-technical-deep-dive/#:~:text=With%20a%20valid%20domain%20user,quickly%20found%20its%20fastest%20route)[[12]](https://horizon3.ai/intelligence/blogs/nodezero-vs-goad-technical-deep-dive/#:~:text=Image). It can forge machine accounts or constrained-delegation entries (RBCD attacks, e.g. CVE-2022-33679, CVE-2022-26923) to escalate privileges[[13]](https://docs.horizon3.ai/knowledge_base/attack_configurations/#:~:text=Option%20Description%20Risk%20Active%20Directory,Deletion%20of%20machine). When a domain credential is obtained, NodeZero will use it to scan SMB shares, use pass-the-hash, or attempt BloodHound-style attacks. All exploited AD weak points are combined into attack paths. In testing, NodeZero has uncovered credentials stored in user descriptions or scripts on AD, then used them to pivot through the domain[[14]](https://horizon3.ai/intelligence/blogs/nodezero-vs-goad-technical-deep-dive/#:~:text=Username%20and%20computer%20name%20lists,in%20his%20Active%20Directory%20description)[[15]](https://horizon3.ai/intelligence/blogs/nodezero-vs-goad-technical-deep-dive/#:~:text=As%20part%20of%20this%20enumeration%2C,roast%20attack%2C%20as%20shown%20below). NodeZero also searches files and registry keys for secrets, and can exploit external services (e.g. log4j/Log4Shell) to pivot into cloud environments like AWS or Azure[[16]](https://horizon3.ai/nodezero/internal-pentesting/#:~:text=In%20this%20real,in%20the%20client%E2%80%99s%20AWS%20environment)[[17]](https://horizon3.ai/attack-research/n0-attack-paths/nodezero-apt-azure-password-spray-leads-to-business-email-compromise/#:~:text=Here%20are%20the%20steps%20NodeZero,get%20to%20business%20email%20compromise).

NodeZero’s modules are continually updated; it can check emerging flaws such as the recent “RegreSSHion” (CVE-2024-6387)[[18]](https://docs.horizon3.ai/knowledge_base/attack_configurations/#:~:text=Option%20Description%20Risk%20RegreSSHion%20check,none). Every module can run safely by default (most are non-destructive), and reports its risk level. It aligns findings with MITRE ATT&CK tactics and includes new N-day tests as they become available[[19]](https://horizon3.ai/nodezero/internal-pentesting/#:~:text=Weaknesses%20uncovered%20in%20your%20internal,pentest%20may%20include). After exploitation, NodeZero tries to chain further attacks (e.g. moving from one compromised host to the next) without human scripts, emulating an attacker’s lateral movement[[20]](https://horizon3.ai/nodezero/internal-pentesting/#:~:text=Find%20and%20eliminate%20the%20weaknesses,don%E2%80%99t%20involve%20exploiting%20any%20CVEs).

* **Examples of exploit modules:** SMBv1 exploits (EternalRomance, MS17-010 variants), BlueKeep RDP exploit, Cisco router install protocol, Oracle WebLogic, Zoho / ManageEngine RCEs, etc.[[8]](https://docs.horizon3.ai/knowledge_base/attack_configurations/#:~:text=Option%20Description%20Risk%20Bluekeep%20%28CVE,high).
* **Default credentials:** It tests common default/usernames against services (FTP, SSH, databases, SNMP, web panels, VMware, etc.)[[10]](https://docs.horizon3.ai/knowledge_base/attack_configurations/#:~:text=Option%20Description%20Risk%20FTP%20Enables,default%20credentials%20against%20PostgreSQL%20databases).
* **AD-specific exploits:** Kerberos attacks (Kerberoast, UnPAC), AS-REP roast, AD CS misconfig exploitation, machine-account takeover (RBCD)[[11]](https://horizon3.ai/intelligence/blogs/nodezero-vs-goad-technical-deep-dive/#:~:text=With%20a%20valid%20domain%20user,quickly%20found%20its%20fastest%20route)[[13]](https://docs.horizon3.ai/knowledge_base/attack_configurations/#:~:text=Option%20Description%20Risk%20Active%20Directory,Deletion%20of%20machine).
* **Cloud and identity:** AWS/Azure token abuse, phishing simulations, and Azure AD password spraying (NodeZero has an “MS Entra Credential Pivoting” and “Azure AD Password Spray” module[[21]](https://docs.horizon3.ai/knowledge_base/attack_configurations/#:~:text=MS%20Entra%20,moderate); e.g. it autonomously enumerated a target’s Azure tenant, used OSINT to find user names, and safely sprayed passwords until one succeeded[[17]](https://horizon3.ai/attack-research/n0-attack-paths/nodezero-apt-azure-password-spray-leads-to-business-email-compromise/#:~:text=Here%20are%20the%20steps%20NodeZero,get%20to%20business%20email%20compromise)[[22]](https://horizon3.ai/attack-research/n0-attack-paths/nodezero-apt-azure-password-spray-leads-to-business-email-compromise/#:~:text=Takeaways)).
* **Man-in-the-Middle:** Captures credentials via LLMNR/NetBIOS poisoning or SMB relaying when in-scope[[6]](https://docs.horizon3.ai/knowledge_base/attack_configurations/#:~:text=match%20at%20L539%20Option%20Description,none).

## Credential Theft & RAT-Based Post-Exploitation

NodeZero treats valid credentials as first-class entry points. Whenever it obtains an account with **local administrator** rights on a host, it uses that credential to deploy a built-in RAT (its custom Remote Access Tool) over SMB, WMI, WinRM, SSH/SFTP, or even database features[[23]](https://horizon3.ai/attack-research/attack-blogs/what-7000-nodezero-rat-attempts-show-us-about-cyber-security/#:~:text=Credential%20Based%20Opportunities)[[24]](https://horizon3.ai/attack-research/attack-blogs/what-7000-nodezero-rat-attempts-show-us-about-cyber-security/#:~:text=On%20Linux%2C%20if%20we%20find,probability%20of%20landing%20the%20RAT). In one recent study, over **7,600 RAT install attempts**, 96.6% were via credentials (SMB, SSH, Postgres, etc.) rather than direct exploits[[25]](https://horizon3.ai/attack-research/attack-blogs/what-7000-nodezero-rat-attempts-show-us-about-cyber-security/#:~:text=We%20gathered%20some%20data%20from,attempts%20use%20credentials%2C%20not%20vulnerabilities)【46】. The most-used NodeZero RAT modules are smb\_exec (Windows), ssh\_exec (Linux), and postgresql\_rce (for PostgreSQL)[[26]](https://horizon3.ai/attack-research/attack-blogs/what-7000-nodezero-rat-attempts-show-us-about-cyber-security/#:~:text=If%20we%20break%20the%20data,installation%20attempts%20by%20a%20landslide)【47】. (This emphasizes that NodeZero prioritizes credential-based intrusion, which tends to evade EDR systems more easily than noisy exploits[[27]](https://horizon3.ai/attack-research/attack-blogs/what-7000-nodezero-rat-attempts-show-us-about-cyber-security/#:~:text=To%20add%20insult%20to%20injury%2C,less%20scrutiny%20from%20security%20systems).)

Once the RAT is running on an endpoint, NodeZero shifts to post-exploitation mode. It immediately performs telemetry (enumerating users, processes, network connections, installed software) and then runs a battery of credential-dumping and data-gathering modules[[28]](https://horizon3.ai/attack-research/attack-blogs/what-7000-nodezero-rat-attempts-show-us-about-cyber-security/#:~:text=Stage%202%3A%20Autonomous%20Post)[[29]](https://horizon3.ai/attack-research/attack-blogs/what-7000-nodezero-rat-attempts-show-us-about-cyber-security/#:~:text=If%20NodeZero%20discovers%20that%20a,and%20particular%20user%20DPAPI%20keys). On Windows, it dumps LSASS, LSA secrets, SAM database, and DPAPI master keys, and can extract cached and live credentials from memory[[30]](https://horizon3.ai/intelligence/blogs/nodezero-vs-goad-technical-deep-dive/#:~:text=Next%2C%20NodeZero%20leveraged%20its%20new,exploitation%20phase). It searches files, browser data, and PKI stores for secrets. For example, NodeZero can remotely retrieve a user’s Chrome cookies and corresponding DPAPI keys, decrypt them offline, and use any valid session cookies (e.g. Okta, Slack) via the RAT to break into those services without triggering “impossible travel” alerts[[29]](https://horizon3.ai/attack-research/attack-blogs/what-7000-nodezero-rat-attempts-show-us-about-cyber-security/#:~:text=If%20NodeZero%20discovers%20that%20a,and%20particular%20user%20DPAPI%20keys)[[31]](https://horizon3.ai/attack-research/attack-blogs/what-7000-nodezero-rat-attempts-show-us-about-cyber-security/#:~:text=Next%2C%20NodeZero%20will%20then%20analyze,where%20NodeZero%20found%20the%20cookie). In tests, NodeZero achieved complete domain compromise by chaining techniques: dumping a user’s hash with the RAT, then passing it to a domain controller to obtain a Golden Ticket[[32]](https://horizon3.ai/intelligence/blogs/nodezero-vs-goad-technical-deep-dive/#:~:text=also%20searched%20the%20file%20system,stark)[[12]](https://horizon3.ai/intelligence/blogs/nodezero-vs-goad-technical-deep-dive/#:~:text=Image).

*Figure: NodeZero RAT deployment attempts by method. In monitored tests (>7,600 attempts), 96.6% used credentials (e.g. SMB/SSH) rather than direct exploits*[*[25]*](https://horizon3.ai/attack-research/attack-blogs/what-7000-nodezero-rat-attempts-show-us-about-cyber-security/#:~:text=We%20gathered%20some%20data%20from,attempts%20use%20credentials%2C%20not%20vulnerabilities)*.*

*Figure: Top NodeZero RAT modules by usage. smb\_exec (Windows) and ssh\_exec (SSH on Linux) dominate, reflecting NodeZero’s credential-based strategy*[*[26]*](https://horizon3.ai/attack-research/attack-blogs/what-7000-nodezero-rat-attempts-show-us-about-cyber-security/#:~:text=If%20we%20break%20the%20data,installation%20attempts%20by%20a%20landslide)*.*

Key automated post-exploitation actions include:  
- **Process and memory forensics:** Dumping LSASS, LSA, SAM, DPAPI keys and open sockets.  
- **Token and credential reuse:** Extracting access tokens or NTLM hashes to pass-the-hash/steal-OAuth, then lateral movement[[32]](https://horizon3.ai/intelligence/blogs/nodezero-vs-goad-technical-deep-dive/#:~:text=also%20searched%20the%20file%20system,stark)[[23]](https://horizon3.ai/attack-research/attack-blogs/what-7000-nodezero-rat-attempts-show-us-about-cyber-security/#:~:text=Credential%20Based%20Opportunities).  
- **Data exfiltration:** Stealing sensitive files, database dumps, email content, etc. (Case study: NodeZero exploited a Log4Shell vulnerability to steal AWS keys and pivot into a customer’s cloud[[16]](https://horizon3.ai/nodezero/internal-pentesting/#:~:text=In%20this%20real,in%20the%20client%E2%80%99s%20AWS%20environment).)  
- **Target-specific exploits:** Using gathered creds on other services (e.g. Windows RDP, SSH, SQL) to plant the RAT or open new sessions.

All activity is logged in NodeZero’s knowledge graph and used to automatically find the next steps, without human intervention. In practice, NodeZero can complete a full post-exploitation payload suite in minutes: median time to run its core RAT modules is ~3 minutes[[33]](https://horizon3.ai/attack-research/attack-blogs/what-7000-nodezero-rat-attempts-show-us-about-cyber-security/#:~:text=AI%20Enabled%20Attacker%3A%20Can%20your,Speed%20of%20the%20NodeZero%20RAT).

## Attack Path Chaining & Real-World Proof

Unlike traditional vulnerability scanners, NodeZero **chains multiple weaknesses** just as a human attacker would[[20]](https://horizon3.ai/nodezero/internal-pentesting/#:~:text=Find%20and%20eliminate%20the%20weaknesses,don%E2%80%99t%20involve%20exploiting%20any%20CVEs). It continuously updates its internal graph of assets and credentials, exploring many possible paths in parallel. At every stage NodeZero “safely exploits” the discovered vector (often by merely proving access, not destroying anything) and records proof-of-success. For example, in one scenario NodeZero found a writable SYSVOL script with a password, used it to log into a server, then deployed the RAT to steal privileged domain credentials – all autonomously[[14]](https://horizon3.ai/intelligence/blogs/nodezero-vs-goad-technical-deep-dive/#:~:text=Username%20and%20computer%20name%20lists,in%20his%20Active%20Directory%20description)[[34]](https://horizon3.ai/intelligence/blogs/nodezero-vs-goad-technical-deep-dive/#:~:text=Next%2C%20NodeZero%20leveraged%20its%20new,exploitation%20phase). Attack paths (and their risk) are prioritized so defenders know which chain of exploits leads to the most critical impact[[35]](https://horizon3.ai/nodezero/#:~:text=The%20NodeZero%20platform%20prioritizes%20the,know%20what%20to%20fix%20first).

NodeZero provides **real-time visibility** into each pentest’s progress. Its dashboard displays active attack paths and shows “proof” (screenshots, logs, or C2 callbacks) of each exploit as it happens[[36]](https://horizon3.ai/nodezero/#:~:text=NodeZero%20pivots%20through%20your%20network%2C,time%20view)[[37]](https://horizon3.ai/nodezero/#:~:text=Understand%20the%20Impact). When a pentest finishes, NodeZero generates a detailed report: it gives step-by-step attack chain summaries, the exact vulnerabilities or credentials exploited, and evidence of access. Crucially, NodeZero’s risk scoring is based on *actual exploitability* rather than static CVEs[[38]](https://horizon3.ai/nodezero/internal-pentesting/#:~:text=Goes%20far%20beyond%20CVEs). For each finding it provides remediation guidance. After patches or configuration changes are applied, NodeZero can be re-run (“quick verify”) to confirm the fix.

In short, NodeZero performs **“find-exploit-proof-remediate-verify”** cycles autonomously. It uses real attack techniques rather than theoretical checks, so every reported weakness has been validated by an actual (benign) exploit attempt. For example, in an external test NodeZero enumerated Azure AD tenants, sprayed passwords, and then accessed a test user’s mailbox in under 10 minutes – with complete logs of each step[[17]](https://horizon3.ai/attack-research/n0-attack-paths/nodezero-apt-azure-password-spray-leads-to-business-email-compromise/#:~:text=Here%20are%20the%20steps%20NodeZero,get%20to%20business%20email%20compromise)[[22]](https://horizon3.ai/attack-research/n0-attack-paths/nodezero-apt-azure-password-spray-leads-to-business-email-compromise/#:~:text=Takeaways). Such attack-story case studies (see GOAD, Azure AD, etc.) show NodeZero’s real-world efficacy.

## Stealth and Evasion Tactics

NodeZero is designed to minimize detection. By default it avoids high-risk actions (flagged in the UI) and runs benign exploits whenever possible. A key insight is that **credentialed access is stealthier**: as Horizon3 found, installing a RAT using legitimate creds often slips past EDRs, whereas triggering a kernel exploit is noisy[[39]](https://horizon3.ai/attack-research/attack-blogs/what-7000-nodezero-rat-attempts-show-us-about-cyber-security/#:~:text=Security%20systems%20face%20this%20same,%E2%80%9Cright%E2%80%9D%20way%20to%20get%20in). Thus NodeZero prioritizes credential-based vectors (SMB/SSH/etc.) over noisy zero-days[[27]](https://horizon3.ai/attack-research/attack-blogs/what-7000-nodezero-rat-attempts-show-us-about-cyber-security/#:~:text=To%20add%20insult%20to%20injury%2C,less%20scrutiny%20from%20security%20systems). It also limits noisy actions: e.g. its Azure password-spray module tries only 2–3 guesses per account per hour to avoid lockouts[[40]](https://horizon3.ai/attack-research/n0-attack-paths/nodezero-apt-azure-password-spray-leads-to-business-email-compromise/#:~:text=Image).

NodeZero also offers adjustable stealth settings. Scanning speed can be slowed; man-in-the-middle attacks can be limited to in-scope segments; and it never deploys permanent agents on targets (except the ephemeral RAT during the test)[[41]](https://sourceforge.net/software/compare/Cobalt-Strike-vs-Horizon3.ai-vs-Pentera/#:~:text=Horizon3,Proactively%20identify%20and%20remediate)[[42]](https://horizon3.ai/nodezero/#:~:text=Quickly%20Set%20Up%20a%20Host). Its “safe by default” configuration means it will not crash production services – users are advised that only a few exploits (like BlueKeep or MS08-067) carry moderate crash risk[[8]](https://docs.horizon3.ai/knowledge_base/attack_configurations/#:~:text=Option%20Description%20Risk%20Bluekeep%20%28CVE,high)[[43]](https://docs.horizon3.ai/knowledge_base/attack_configurations/#:~:text=Server%20Service%20Vulnerability%20%28MS08,high). Because NodeZero runs in a disposable VM/container, it leaves no lasting footprint. In practice this means NodeZero can be run **continuously** (hourly/daily) against production networks without causing downtime.

## Comparison to Other Platforms

Compared to manual red-team tools and BAS platforms, NodeZero is unique in its full-stack automation:

* **Cobalt Strike (Fortra):** A human-driven adversary-simulation toolkit focused on post-exploitation. It provides a beacon/C2 and Malleable C2 scripts, but *no* built-in network recon or automated chaining. Cobalt Strike requires manual human operators and typically needs an initial foothold (e.g. phishing) to be planted. In contrast, NodeZero automates scanning, exploitation and lateral movement end-to-end[[44]](https://horizon3.ai/attack-research/attack-blogs/what-7000-nodezero-rat-attempts-show-us-about-cyber-security/#:~:text=Part%20of%20a%20manual%20penetration,decided%20to%20write%20our%20own).
* **Metasploit Pro:** An enterprise pentesting suite combining Metasploit exploits, scanners and pivoting. It can automate credentialed scans and module runs, but it is largely sequential (exploits are queued and require user control). It lacks NodeZero’s autonomous graph-driven logic and often requires scripting or manual intervention to link exploits and dumps. NodeZero goes beyond Metasploit by autonomously chaining credentials and using outcomes to select next steps.
* **Pentera (Pcysys):** An automated validation platform (Breach-and-Attack Simulation). Pentera is agentless but is primarily limited to internal network vulnerability checks. Notably, Pentera **does not perform credential harvesting or exploitation** during scans[[45]](https://static.rainfocus.com/rsac/us23/exh/16394317879160012aNt/exhibitorboothresource/Horizon3_BAS-Pentera-Comparison_1675982075122001fVxJ.pdf#:~:text=Horizon3,end%20gaming%20laptop). Pentera can run some exploit modules, but Horizon3 points out it **cannot attack from outside** or cloud environments, and typically requires a dedicated on-premises appliance (often a specialized laptop)[[46]](https://static.rainfocus.com/rsac/us23/exh/16394317879160012aNt/exhibitorboothresource/Horizon3_BAS-Pentera-Comparison_1675982075122001fVxJ.pdf#:~:text=Horizon3.ai%20does%20NOT%20require%20on,many%20perspectives%20%E2%80%93%20internal%2C%20external). By contrast, NodeZero can attack internal and external scopes (including home/remote subnets), pivot into cloud workloads, and is designed to **safely run against production** networks[[47]](https://static.rainfocus.com/rsac/us23/exh/16394317879160012aNt/exhibitorboothresource/Horizon3_BAS-Pentera-Comparison_1675982075122001fVxJ.pdf#:~:text=Horizon3,to%20run%20against%20production%20systems). Both platforms align reports to MITRE ATT&CK, but NodeZero also auto-generates consultant-style pentest reports with detailed fix actions[[48]](https://static.rainfocus.com/rsac/us23/exh/16394317879160012aNt/exhibitorboothresource/Horizon3_BAS-Pentera-Comparison_1675982075122001fVxJ.pdf#:~:text=Reports%20results%20aligned%20to%20MITRE,to%20common%20frameworks%20like%20MITRE).
* **Breach-and-Attack Simulation (BAS) Tools** (e.g. AttackIQ, SafeBreach): These tools simulate adversary techniques to test defenses, but typically *do not* execute real exploits on target systems. They focus on purple-team drills and detection validation. For example, BAS tools will emulate credential stuffing but won’t actually dump LSASS or hijack tokens. NodeZero, in contrast, uses *real exploits* and credential theft to *prove* risk. (As one Horizon3 comparison slide notes, NodeZero **can harvest, crack, and reuse credentials**, while most BAS tools cannot[[49]](https://static.rainfocus.com/rsac/us23/exh/16394317879160012aNt/exhibitorboothresource/Horizon3_BAS-Pentera-Comparison_1675982075122001fVxJ.pdf#:~:text=Do%20you%20require%20credentialed%2C%20persistent,sensitive%20data%20and%20elevate%20privileges).) Additionally, NodeZero requires no custom script development – its attack library is built-in and updated by Horizon3 – whereas BAS workflows often rely on scripted playbooks.

In summary, NodeZero blends automated attack (like Pentera/BAS) with real exploitation (like Metasploit) and autonomous chaining (beyond Cobalt Strike). Its **agentless, credential-driven approach** and continuous operation differentiate it from legacy pentest tools. (See comparison table below.)

| Feature / Capability | Horizon3.ai NodeZero | Cobalt Strike (Fortra) | Metasploit Pro (Rapid7) | Pentera (Pcysys) | BAS Platforms (AttackIQ, etc.) |
| --- | --- | --- | --- | --- | --- |
| **Scope (Internal/External/Cloud)** | Internal, External, Cloud (on-prem or SaaS)[[47]](https://static.rainfocus.com/rsac/us23/exh/16394317879160012aNt/exhibitorboothresource/Horizon3_BAS-Pentera-Comparison_1675982075122001fVxJ.pdf#:~:text=Horizon3,to%20run%20against%20production%20systems) | Typically internal (after phish) | Internal (on-prem install) | Internal only[[47]](https://static.rainfocus.com/rsac/us23/exh/16394317879160012aNt/exhibitorboothresource/Horizon3_BAS-Pentera-Comparison_1675982075122001fVxJ.pdf#:~:text=Horizon3,to%20run%20against%20production%20systems) | Internal & limited external (simulated) |
| **Agents Required** | **No persistent agents;** uses ephemeral RAT | C2 agent (Beacon) on compromise | Metasploit payload (Meterpreter) | Agentless (requires “Pentera box” on prem)[[50]](https://static.rainfocus.com/rsac/us23/exh/16394317879160012aNt/exhibitorboothresource/Horizon3_BAS-Pentera-Comparison_1675982075122001fVxJ.pdf#:~:text=Horizon3.ai%20does%20NOT%20require%20on,and%20the%20user%20must%20maintain) | Agentless (but often requires test agents for detection testing) |
| **Credential Exploitation** | Yes – collects, cracks, reuses credentials[[45]](https://static.rainfocus.com/rsac/us23/exh/16394317879160012aNt/exhibitorboothresource/Horizon3_BAS-Pentera-Comparison_1675982075122001fVxJ.pdf#:~:text=Horizon3,end%20gaming%20laptop) | Supports credential harvesting via sessions (manual) | Yes, supports credentialed pivoting | **No credential attacks**[[45]](https://static.rainfocus.com/rsac/us23/exh/16394317879160012aNt/exhibitorboothresource/Horizon3_BAS-Pentera-Comparison_1675982075122001fVxJ.pdf#:~:text=Horizon3,end%20gaming%20laptop) | Limited – usually simulated or post-exploitation token stuffing |
| **Automated Chaining** | Full – autonomously chains exploits, lateral moves[[20]](https://horizon3.ai/nodezero/internal-pentesting/#:~:text=Find%20and%20eliminate%20the%20weaknesses,don%E2%80%99t%20involve%20exploiting%20any%20CVEs) | No – manual red-team choice of steps | Partial – scripts/manual | No – scans are one-shot per target, no adaptive chaining | No – simulations follow scripted scenarios |
| **Real Exploits vs Simulated** | Real exploits & payloads on live systems | Real payloads (requires initial foothold) | Real exploits against discovered vulns | Real exploits (but only internal, do-no-harm) | Simulated tactics (no destructive payloads) |
| **Stealth / Prod Safe** | Safe-by-default config; can run in production[[47]](https://static.rainfocus.com/rsac/us23/exh/16394317879160012aNt/exhibitorboothresource/Horizon3_BAS-Pentera-Comparison_1675982075122001fVxJ.pdf#:~:text=Horizon3,to%20run%20against%20production%20systems) | Covert by design (malleable C2), but runs as intrusive process | Can crash targets (heavy exploits) | Can disrupt services; Pentera cautions on prod | Non-disruptive simulation |
| **Reporting** | Automated pentest-style reports with fix actions; MITRE-aligned[[48]](https://static.rainfocus.com/rsac/us23/exh/16394317879160012aNt/exhibitorboothresource/Horizon3_BAS-Pentera-Comparison_1675982075122001fVxJ.pdf#:~:text=Reports%20results%20aligned%20to%20MITRE,to%20common%20frameworks%20like%20MITRE) | Focus on engagement reports; no automated remediation guidance | Automated and manual reports; CVSS-based | Automated risk reports (MITRE framework) | Adversary behavior and detection gap reports |
| **Continuous/Repeat Testing** | Yes – designed for scheduled/continuous use[[51]](https://horizon3.ai/nodezero/#:~:text=You%20can%20set%20up%20and,first%20NodeZero%20pentest%20in%20minutes) | No – used ad-hoc per engagement | Ad-hoc use | Typically periodic scans | Continuous assessment (simulated attacks) |
| **API / Integrations** | Rich GraphQL API, CLI (h3-cli), Splunk app[[52]](https://docs.horizon3.ai/api/#:~:text=Horizon3,automation%2C%20integration%2C%20and%20advanced%20workflows)[[53]](https://github.com/horizon3ai/h3-cli#:~:text=h3,the%20API%20allows%20you%20to) | Limited scripting (Aggressor scripts); no central API | REST/DB for sharing reports | REST API for orchestration | APIs available (focus on SIEM/SOAR integration) |
| **Primary Use Cases** | Automated pentesting, attack path discovery, risk prioritization[[36]](https://horizon3.ai/nodezero/#:~:text=NodeZero%20pivots%20through%20your%20network%2C,time%20view) | Red-team ops, social engineering, C2 emulation | Manual/automated pentesting and exploits | Enterprise security validation | Purple-team drill, SOC validation |

*Table: Comparison of NodeZero vs. common pentesting and BAS tools (features/principles). NodeZero uniquely offers fully autonomous, credential-harvesting attack chains with proof, whereas others are either manual red-team tools or simulation platforms*[*[54]*](https://static.rainfocus.com/rsac/us23/exh/16394317879160012aNt/exhibitorboothresource/Horizon3_BAS-Pentera-Comparison_1675982075122001fVxJ.pdf#:~:text=Horizon3,and%20the%20user%20must%20maintain)[*[38]*](https://horizon3.ai/nodezero/internal-pentesting/#:~:text=Goes%20far%20beyond%20CVEs)*.*

## Integrations and Automation

NodeZero supports extensible integration into security workflows. Horizon3 exposes a full **GraphQL API** for NodeZero: users can programmatically launch pentests, set scopes, and retrieve results in JSON[[52]](https://docs.horizon3.ai/api/#:~:text=Horizon3,automation%2C%20integration%2C%20and%20advanced%20workflows). A command-line tool h3-cli wraps this API: it can schedule tests, install the NodeZero Runner, monitor progress, and download reports[[53]](https://github.com/horizon3ai/h3-cli#:~:text=h3,the%20API%20allows%20you%20to). This allows embedding NodeZero into CI/CD or ticketing pipelines. Token-based auth and role-based API keys ensure secure automation[[52]](https://docs.horizon3.ai/api/#:~:text=Horizon3,automation%2C%20integration%2C%20and%20advanced%20workflows).

There are prebuilt integrations: for example, a **Splunk App** ingests NodeZero findings into Splunk for correlation with logs[[55]](https://docs.horizon3.ai/downloads/splunk_app/#:~:text=Splunk%20App%20,attacker%27s%20perspective%20with%20existing). The NodeZero Knowledge Graph can feed dashboards or trigger SOAR playbooks via webhooks or the API. Horizon3 also offers an **MCP Server** which bridges LLM/AI agents to the NodeZero API, enabling natural-language orchestration of pentests[[56]](https://docs.horizon3.ai/portal/features/mcp/#:~:text=The%20NodeZero%20MCP%20,FFV%29%20cycles)[[57]](https://docs.horizon3.ai/portal/features/mcp/#:~:text=Use%20the%20MCP%20Server%20to,intelligence%20in%20your%20NodeZero%20workflows). In practice, any application that can make REST/GraphQL calls can initiate NodeZero operations and consume its JSON output.

NodeZero itself can be deployed flexibly. Internal tests run a free Docker/OVA host in the network[[42]](https://horizon3.ai/nodezero/#:~:text=Quickly%20Set%20Up%20a%20Host), while external tests run from the Horizon3 cloud. No persistent VPN or credentials need to be pre-provisioned (NodeZero only needs network reachability to in-scope IPs). Horizon3 provides VM images (Linux/Ubuntu) and scripts to spin up the NodeZero Runner host.

## Real-World Usage

NodeZero is in active use by consulting firms and enterprises. For instance, NCC Group now uses NodeZero’s AI-driven platform to augment its penetration testing services, achieving “autonomous and hybrid penetration tests” with faster vulnerability verification and remediation prioritization[[58]](https://www.nccgroup.com/newsroom/ncc-group-adds-horizon3ai-s-nodezero-platform-to-network-penetration-testing-capabilities/#:~:text=Kevin%20Brown%2C%20Chief%20Operating%20Officer%2C,challenging%20vulnerabilities%20and%20conducting%20research)[[59]](https://www.nccgroup.com/newsroom/ncc-group-adds-horizon3ai-s-nodezero-platform-to-network-penetration-testing-capabilities/#:~:text=Snehal%20Antani%2C%20CEO%20and%20Co,%E2%80%9D). In customer engagements, NodeZero has autonomously discovered exposed AWS keys via Log4Shell, achieved domain-wide compromise via forged Golden Tickets, and performed external Azure AD password-spray attacks leading to Business Email Compromise – all with zero human intervention[[11]](https://horizon3.ai/intelligence/blogs/nodezero-vs-goad-technical-deep-dive/#:~:text=With%20a%20valid%20domain%20user,quickly%20found%20its%20fastest%20route)[[17]](https://horizon3.ai/attack-research/n0-attack-paths/nodezero-apt-azure-password-spray-leads-to-business-email-compromise/#:~:text=Here%20are%20the%20steps%20NodeZero,get%20to%20business%20email%20compromise). By continuously testing, NodeZero helps organizations “stay ahead of evolving threats” by repeatedly finding and fixing exploitable paths[[59]](https://www.nccgroup.com/newsroom/ncc-group-adds-horizon3ai-s-nodezero-platform-to-network-penetration-testing-capabilities/#:~:text=Snehal%20Antani%2C%20CEO%20and%20Co,%E2%80%9D)[[60]](https://horizon3.ai/nodezero/#:~:text=Use%20detailed%20fix%20guidance%2C%20then,verify%20that%20your%20fixes%20worked).

In summary, Horizon3.ai’s NodeZero platform leverages a comprehensive suite of pentest techniques – from network and AD exploits to credential harvesting, RAT deployment, and data pilfering – in a fully automated framework. It validates each finding with real exploit proof, chains multiple steps like an attacker, and integrates via APIs for custom workflows. This end-to-end automation distinguishes NodeZero from manual toolkits and simulation-only BAS platforms, delivering continuous, evidence-driven vulnerability assessment.

**Sources:** Public Horizon3.ai documentation, blogs, demos and product materials[[1]](https://horizon3.ai/intelligence/blogs/nodezero-vs-goad-technical-deep-dive/#:~:text=The%20first%20few%20minutes%20were,across%20three%20Active%20Directory%20domains)[[8]](https://docs.horizon3.ai/knowledge_base/attack_configurations/#:~:text=Option%20Description%20Risk%20Bluekeep%20%28CVE,high)[[36]](https://horizon3.ai/nodezero/#:~:text=NodeZero%20pivots%20through%20your%20network%2C,time%20view)[[45]](https://static.rainfocus.com/rsac/us23/exh/16394317879160012aNt/exhibitorboothresource/Horizon3_BAS-Pentera-Comparison_1675982075122001fVxJ.pdf#:~:text=Horizon3,end%20gaming%20laptop)[[44]](https://horizon3.ai/attack-research/attack-blogs/what-7000-nodezero-rat-attempts-show-us-about-cyber-security/#:~:text=Part%20of%20a%20manual%20penetration,decided%20to%20write%20our%20own)[[61]](https://horizon3.ai/wp-content/uploads/2025/06/creds_vs_vuln-edited.png#:~:text=)[[62]](https://horizon3.ai/wp-content/uploads/2025/06/top_10_modules-1-edited.png#:~:text=), plus NCC Group press release[[58]](https://www.nccgroup.com/newsroom/ncc-group-adds-horizon3ai-s-nodezero-platform-to-network-penetration-testing-capabilities/#:~:text=Kevin%20Brown%2C%20Chief%20Operating%20Officer%2C,challenging%20vulnerabilities%20and%20conducting%20research). All features are derived from published NodeZero modules and case studies.

[[1]](https://horizon3.ai/intelligence/blogs/nodezero-vs-goad-technical-deep-dive/#:~:text=The%20first%20few%20minutes%20were,across%20three%20Active%20Directory%20domains) [[11]](https://horizon3.ai/intelligence/blogs/nodezero-vs-goad-technical-deep-dive/#:~:text=With%20a%20valid%20domain%20user,quickly%20found%20its%20fastest%20route) [[12]](https://horizon3.ai/intelligence/blogs/nodezero-vs-goad-technical-deep-dive/#:~:text=Image) [[14]](https://horizon3.ai/intelligence/blogs/nodezero-vs-goad-technical-deep-dive/#:~:text=Username%20and%20computer%20name%20lists,in%20his%20Active%20Directory%20description) [[15]](https://horizon3.ai/intelligence/blogs/nodezero-vs-goad-technical-deep-dive/#:~:text=As%20part%20of%20this%20enumeration%2C,roast%20attack%2C%20as%20shown%20below) [[30]](https://horizon3.ai/intelligence/blogs/nodezero-vs-goad-technical-deep-dive/#:~:text=Next%2C%20NodeZero%20leveraged%20its%20new,exploitation%20phase) [[32]](https://horizon3.ai/intelligence/blogs/nodezero-vs-goad-technical-deep-dive/#:~:text=also%20searched%20the%20file%20system,stark) [[34]](https://horizon3.ai/intelligence/blogs/nodezero-vs-goad-technical-deep-dive/#:~:text=Next%2C%20NodeZero%20leveraged%20its%20new,exploitation%20phase) Autonomous NodeZero Breaches GOAD via AD Misconfigurations

<https://horizon3.ai/intelligence/blogs/nodezero-vs-goad-technical-deep-dive/>

[[2]](https://horizon3.ai/nodezero/internal-pentesting/#:~:text=During%20the%20discovery%2Fnetwork%20enumeration%20phase,systems%2C%20web%20servers%2C%20and%20applications) [[7]](https://horizon3.ai/nodezero/internal-pentesting/#:~:text=%2A%20Exposed%20credentials%C2%A0include%C2%A0Man,run%20regular%C2%A0password%20audits%20with%20NodeZero) [[16]](https://horizon3.ai/nodezero/internal-pentesting/#:~:text=In%20this%20real,in%20the%20client%E2%80%99s%20AWS%20environment) [[19]](https://horizon3.ai/nodezero/internal-pentesting/#:~:text=Weaknesses%20uncovered%20in%20your%20internal,pentest%20may%20include) [[20]](https://horizon3.ai/nodezero/internal-pentesting/#:~:text=Find%20and%20eliminate%20the%20weaknesses,don%E2%80%99t%20involve%20exploiting%20any%20CVEs) [[38]](https://horizon3.ai/nodezero/internal-pentesting/#:~:text=Goes%20far%20beyond%20CVEs) Continuous Internal Pentesting | Horizon3.ai

<https://horizon3.ai/nodezero/internal-pentesting/>

[[3]](https://horizon3.ai/glossary-vocab/osint/#:~:text=OSINT%20is%20the%20acronym%20for,just%20as%20an%20attacker%20might) OSINT | Horizon3.ai

<https://horizon3.ai/glossary-vocab/osint/>

[[4]](https://docs.horizon3.ai/portal/deployment_strategy/#:~:text=6) [[5]](https://docs.horizon3.ai/portal/deployment_strategy/#:~:text=Intelligent%20Scope) NodeZero Deployment Strategy - HORIZON3

<https://docs.horizon3.ai/portal/deployment_strategy/>

[[6]](https://docs.horizon3.ai/knowledge_base/attack_configurations/#:~:text=match%20at%20L539%20Option%20Description,none) [[8]](https://docs.horizon3.ai/knowledge_base/attack_configurations/#:~:text=Option%20Description%20Risk%20Bluekeep%20%28CVE,high) [[9]](https://docs.horizon3.ai/knowledge_base/attack_configurations/#:~:text=EternalSynergy%2C%20and%20EternalRomance,based%20buffer%20overflow.%20low) [[10]](https://docs.horizon3.ai/knowledge_base/attack_configurations/#:~:text=Option%20Description%20Risk%20FTP%20Enables,default%20credentials%20against%20PostgreSQL%20databases) [[13]](https://docs.horizon3.ai/knowledge_base/attack_configurations/#:~:text=Option%20Description%20Risk%20Active%20Directory,Deletion%20of%20machine) [[18]](https://docs.horizon3.ai/knowledge_base/attack_configurations/#:~:text=Option%20Description%20Risk%20RegreSSHion%20check,none) [[21]](https://docs.horizon3.ai/knowledge_base/attack_configurations/#:~:text=MS%20Entra%20,moderate) [[43]](https://docs.horizon3.ai/knowledge_base/attack_configurations/#:~:text=Server%20Service%20Vulnerability%20%28MS08,high) Attack Configurations - HORIZON3

<https://docs.horizon3.ai/knowledge_base/attack_configurations/>

[[17]](https://horizon3.ai/attack-research/n0-attack-paths/nodezero-apt-azure-password-spray-leads-to-business-email-compromise/#:~:text=Here%20are%20the%20steps%20NodeZero,get%20to%20business%20email%20compromise) [[22]](https://horizon3.ai/attack-research/n0-attack-paths/nodezero-apt-azure-password-spray-leads-to-business-email-compromise/#:~:text=Takeaways) [[40]](https://horizon3.ai/attack-research/n0-attack-paths/nodezero-apt-azure-password-spray-leads-to-business-email-compromise/#:~:text=Image) NodeZero APT: Azure Password Spray Leads to Business Email Compromise | Horizon3.ai

<https://horizon3.ai/attack-research/n0-attack-paths/nodezero-apt-azure-password-spray-leads-to-business-email-compromise/>

[[23]](https://horizon3.ai/attack-research/attack-blogs/what-7000-nodezero-rat-attempts-show-us-about-cyber-security/#:~:text=Credential%20Based%20Opportunities) [[24]](https://horizon3.ai/attack-research/attack-blogs/what-7000-nodezero-rat-attempts-show-us-about-cyber-security/#:~:text=On%20Linux%2C%20if%20we%20find,probability%20of%20landing%20the%20RAT) [[25]](https://horizon3.ai/attack-research/attack-blogs/what-7000-nodezero-rat-attempts-show-us-about-cyber-security/#:~:text=We%20gathered%20some%20data%20from,attempts%20use%20credentials%2C%20not%20vulnerabilities) [[26]](https://horizon3.ai/attack-research/attack-blogs/what-7000-nodezero-rat-attempts-show-us-about-cyber-security/#:~:text=If%20we%20break%20the%20data,installation%20attempts%20by%20a%20landslide) [[27]](https://horizon3.ai/attack-research/attack-blogs/what-7000-nodezero-rat-attempts-show-us-about-cyber-security/#:~:text=To%20add%20insult%20to%20injury%2C,less%20scrutiny%20from%20security%20systems) [[28]](https://horizon3.ai/attack-research/attack-blogs/what-7000-nodezero-rat-attempts-show-us-about-cyber-security/#:~:text=Stage%202%3A%20Autonomous%20Post) [[29]](https://horizon3.ai/attack-research/attack-blogs/what-7000-nodezero-rat-attempts-show-us-about-cyber-security/#:~:text=If%20NodeZero%20discovers%20that%20a,and%20particular%20user%20DPAPI%20keys) [[31]](https://horizon3.ai/attack-research/attack-blogs/what-7000-nodezero-rat-attempts-show-us-about-cyber-security/#:~:text=Next%2C%20NodeZero%20will%20then%20analyze,where%20NodeZero%20found%20the%20cookie) [[33]](https://horizon3.ai/attack-research/attack-blogs/what-7000-nodezero-rat-attempts-show-us-about-cyber-security/#:~:text=AI%20Enabled%20Attacker%3A%20Can%20your,Speed%20of%20the%20NodeZero%20RAT) [[39]](https://horizon3.ai/attack-research/attack-blogs/what-7000-nodezero-rat-attempts-show-us-about-cyber-security/#:~:text=Security%20systems%20face%20this%20same,%E2%80%9Cright%E2%80%9D%20way%20to%20get%20in) [[44]](https://horizon3.ai/attack-research/attack-blogs/what-7000-nodezero-rat-attempts-show-us-about-cyber-security/#:~:text=Part%20of%20a%20manual%20penetration,decided%20to%20write%20our%20own) Insights from 7,000+ NodeZero RAT Deployments | Horizon3.ai

<https://horizon3.ai/attack-research/attack-blogs/what-7000-nodezero-rat-attempts-show-us-about-cyber-security/>

[[35]](https://horizon3.ai/nodezero/#:~:text=The%20NodeZero%20platform%20prioritizes%20the,know%20what%20to%20fix%20first) [[36]](https://horizon3.ai/nodezero/#:~:text=NodeZero%20pivots%20through%20your%20network%2C,time%20view) [[37]](https://horizon3.ai/nodezero/#:~:text=Understand%20the%20Impact) [[42]](https://horizon3.ai/nodezero/#:~:text=Quickly%20Set%20Up%20a%20Host) [[51]](https://horizon3.ai/nodezero/#:~:text=You%20can%20set%20up%20and,first%20NodeZero%20pentest%20in%20minutes) [[60]](https://horizon3.ai/nodezero/#:~:text=Use%20detailed%20fix%20guidance%2C%20then,verify%20that%20your%20fixes%20worked) The NodeZero Platform | Horizon3.ai

<https://horizon3.ai/nodezero/>

[[41]](https://sourceforge.net/software/compare/Cobalt-Strike-vs-Horizon3.ai-vs-Pentera/#:~:text=Horizon3,Proactively%20identify%20and%20remediate) Cobalt Strike vs. NodeZero by Horizon3.ai vs. Pentera Comparison

<https://sourceforge.net/software/compare/Cobalt-Strike-vs-Horizon3.ai-vs-Pentera/>

[[45]](https://static.rainfocus.com/rsac/us23/exh/16394317879160012aNt/exhibitorboothresource/Horizon3_BAS-Pentera-Comparison_1675982075122001fVxJ.pdf#:~:text=Horizon3,end%20gaming%20laptop) [[46]](https://static.rainfocus.com/rsac/us23/exh/16394317879160012aNt/exhibitorboothresource/Horizon3_BAS-Pentera-Comparison_1675982075122001fVxJ.pdf#:~:text=Horizon3.ai%20does%20NOT%20require%20on,many%20perspectives%20%E2%80%93%20internal%2C%20external) [[47]](https://static.rainfocus.com/rsac/us23/exh/16394317879160012aNt/exhibitorboothresource/Horizon3_BAS-Pentera-Comparison_1675982075122001fVxJ.pdf#:~:text=Horizon3,to%20run%20against%20production%20systems) [[48]](https://static.rainfocus.com/rsac/us23/exh/16394317879160012aNt/exhibitorboothresource/Horizon3_BAS-Pentera-Comparison_1675982075122001fVxJ.pdf#:~:text=Reports%20results%20aligned%20to%20MITRE,to%20common%20frameworks%20like%20MITRE) [[49]](https://static.rainfocus.com/rsac/us23/exh/16394317879160012aNt/exhibitorboothresource/Horizon3_BAS-Pentera-Comparison_1675982075122001fVxJ.pdf#:~:text=Do%20you%20require%20credentialed%2C%20persistent,sensitive%20data%20and%20elevate%20privileges) [[50]](https://static.rainfocus.com/rsac/us23/exh/16394317879160012aNt/exhibitorboothresource/Horizon3_BAS-Pentera-Comparison_1675982075122001fVxJ.pdf#:~:text=Horizon3.ai%20does%20NOT%20require%20on,and%20the%20user%20must%20maintain) [[54]](https://static.rainfocus.com/rsac/us23/exh/16394317879160012aNt/exhibitorboothresource/Horizon3_BAS-Pentera-Comparison_1675982075122001fVxJ.pdf#:~:text=Horizon3,and%20the%20user%20must%20maintain) static.rainfocus.com

<https://static.rainfocus.com/rsac/us23/exh/16394317879160012aNt/exhibitorboothresource/Horizon3_BAS-Pentera-Comparison_1675982075122001fVxJ.pdf>

[[52]](https://docs.horizon3.ai/api/#:~:text=Horizon3,automation%2C%20integration%2C%20and%20advanced%20workflows) NodeZero API - HORIZON3

<https://docs.horizon3.ai/api/>

[[53]](https://github.com/horizon3ai/h3-cli#:~:text=h3,the%20API%20allows%20you%20to) GitHub - horizon3ai/h3-cli: CLI tool for the Horizon3.ai API

<https://github.com/horizon3ai/h3-cli>

[[55]](https://docs.horizon3.ai/downloads/splunk_app/#:~:text=Splunk%20App%20,attacker%27s%20perspective%20with%20existing) Splunk App - Horizon3.ai documentation

<https://docs.horizon3.ai/downloads/splunk_app/>

[[56]](https://docs.horizon3.ai/portal/features/mcp/#:~:text=The%20NodeZero%20MCP%20,FFV%29%20cycles) [[57]](https://docs.horizon3.ai/portal/features/mcp/#:~:text=Use%20the%20MCP%20Server%20to,intelligence%20in%20your%20NodeZero%20workflows) MCP Server - HORIZON3

<https://docs.horizon3.ai/portal/features/mcp/>

[[58]](https://www.nccgroup.com/newsroom/ncc-group-adds-horizon3ai-s-nodezero-platform-to-network-penetration-testing-capabilities/#:~:text=Kevin%20Brown%2C%20Chief%20Operating%20Officer%2C,challenging%20vulnerabilities%20and%20conducting%20research) [[59]](https://www.nccgroup.com/newsroom/ncc-group-adds-horizon3ai-s-nodezero-platform-to-network-penetration-testing-capabilities/#:~:text=Snehal%20Antani%2C%20CEO%20and%20Co,%E2%80%9D) NCC Group adds Horizon3.ai’s NodeZero® platform to network penetration testing capabilities | NCC Group

<https://www.nccgroup.com/newsroom/ncc-group-adds-horizon3ai-s-nodezero-platform-to-network-penetration-testing-capabilities/>

[[61]](https://horizon3.ai/wp-content/uploads/2025/06/creds_vs_vuln-edited.png#:~:text=) horizon3.ai

<https://horizon3.ai/wp-content/uploads/2025/06/creds_vs_vuln-edited.png>

[[62]](https://horizon3.ai/wp-content/uploads/2025/06/top_10_modules-1-edited.png#:~:text=) horizon3.ai

<https://horizon3.ai/wp-content/uploads/2025/06/top_10_modules-1-edited.png>