Third-Party Landing Zone (3PLZ) RFP Overview

Michael J. Martin

Network Security Engineering

November 2025

CVS Health

Contents

CVS Health

Third-Party Landing Zone Overview

Summary This RFP outlines a transformative connectivity solution designed to replace legacy models with secure, scalable, and self-service capabilities for streamlined third-party interactions.

Current State

Distributed, Non-Standard Legacy Models

- Lack of visibility, isolation, auditability, and scalability.
- Manual processes introduce inefficiencies and increase risk.

Inefficient Manual Processes

- Lack of standard security controls based on multiple, manual paths.
- Amplified challenges due to growing third-party interactions.

Increased Risk Exposure

- Evolving regulatory requirements and sophisticated cyber threats.
- Lack of standardized design patterns with embedded security controls.

Future State

Modern Service-Oriented Capabilities

- Transition to secure, repeatable, and compliant solutions.
- Improved isolation, auditability, and scalability.

Automated Processes and Self-Service

- Empower business units through self-service onboarding.
- Automation enhances operational efficiency and standard deployment.

End-to-End Governance and Compliance

- Standardized, automated approach for least privilege and segmentation.
- Enhanced identity-aware controls ensure compliance and operational resilience.

Case for Change To address the security and operational issues of legacy connectivity models, **we need to transition to a modern and automated solution** that **empowers teams with self-service capabilities** while ensuring **standard, secure deployments and scalability.**

Benefits from the Strategic Shift

The implementation of 3PLZ represents a strategic shift from viewing third-party connectivity and provides tangible business benefits.

Self-Service Support for All User Skill Levels

Empowers both technical and non-technical application users to independently manage connectivity through intuitive self-service tools, reducing reliance on manual processes.

Risk Reduction through External Policy Enforcement

Shifting third-party policy enforcement outside the CVS core network reduces risk exposure and enhances security while maintaining seamless connectivity for application teams.

Automation and Scalability

Automated delivery frameworks ensure consistent, repeatable, and scalable connectivity, while embedding compliance and governance controls into every step.

Operational Efficiency and Ownership

The strategic shift fosters enterprise-wide efficiency, reducing manual effort and reinforcing a culture of ownership and accountability for secure connectivity.

Centralized Visibility and Management

Provides a comprehensive view and control of third-party connections, enabling rapid identification and isolation during cyber events to protect CVS from potential threats.

North Star

To enable seamless, secure, and scalable connectivity across the by empowering teams through self-service, reducing risk with external policy enforcement, automating processes, enhancing operational efficiency, and ensuring centralized visibility.

Expected Outcomes

Detailing the high-level overview of the vision for driving the transition of 3PLZ.

Standardized IPSec Connectivity

- Delivers secure VPN connectivity supporting platforms like Cisco and Palo Alto.
- Validated configurations ensure quick deployment with minimal data requirements.
- Supports peer-to-peer routing for isolated communication.

Virtual and Secure Tenant Architecture

- Operates with split-brain proxy model for separate routing domains.
- Aligns with SOC 2, ISO 27001, and NIST standards for compliance.
- Protects enterprise applications from external access.

Granular Access Control

- Service-specific proxies enforce least privilege and segmentation.
- Identity-aware controls ensure context-driven access decisions.
- Delivers auditable, fine-grained access per-partner.

Repeatable, Compliant Connectivity

- Streamlines onboarding with intuitive self-service portal.
- Reduces manual effort while accelerating time-to-market.
- Maintains strict compliance with industry security standards through automation.

Service Pattern Catalogue – Initial Use Cases

The platform is defined around five core service patterns – not just on raw network reachability.

Secure File Transfer

SFTP / SSH / MFT (e.g., IBM Sterling)

Collaboration Access

M365, eShare, Slack via proxied, logged access

Virtual Desktop (VDI)

Citrix / Microsoft VDI via per tenant proxies

Web Applications

Internal HTTPS apps via reverse proxies with mTLS

Web APIs

REST/SOAP APIs with mTLS and token/claim-based controls

Each pattern must support both **Enterprise** and **Internet/SaaS paths**

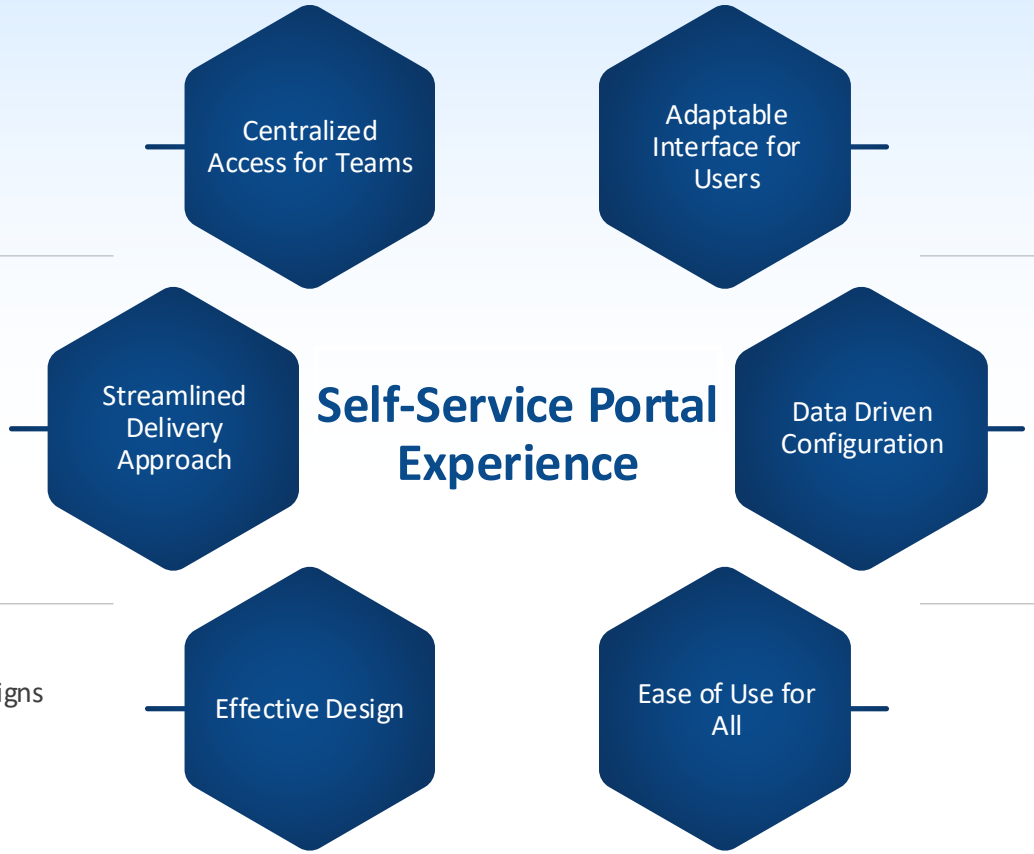
Self-Service Portal Experience

The self-service portal provides a streamlined, user-friendly interface that empowers both technical and non-technical teams to efficiently manage configurations, leveraging methodologies for enhanced operational accessibility.

- Web portal serves as the central access point
- Facilitates capability delivery for application teams
- Streamlines processes for product managers

- Implements the "Shift Left" methodology

- Embraces the KISS principle
- Focuses on easy-to-understand and implement designs
- Minimizes complexity to enhance usability



- Provides both web and API interfaces for flexible access
- Supports ease of adoption for technical and non-technical users
- Ensures accessibility to all user levels

- Utilizes service and pattern-based configurations
- Relies on easily collectable and actionable data
- Enables precise customization of services

- Requires no deep network expertise for users
- Portal offers tested configurations built on defined patterns
- Simplifies process for requestors, enhancing usability

Virtual Landing Zone (VLZ) & Tenant Model

The Virtual Landing Zone (VLZ) & Tenant Model ensures secure and isolated third-party connectivity through dedicated, identity-driven configurations, facilitating robust communication while maintaining no direct access to internal networks.

Virtual Landing Zone (VLZ)

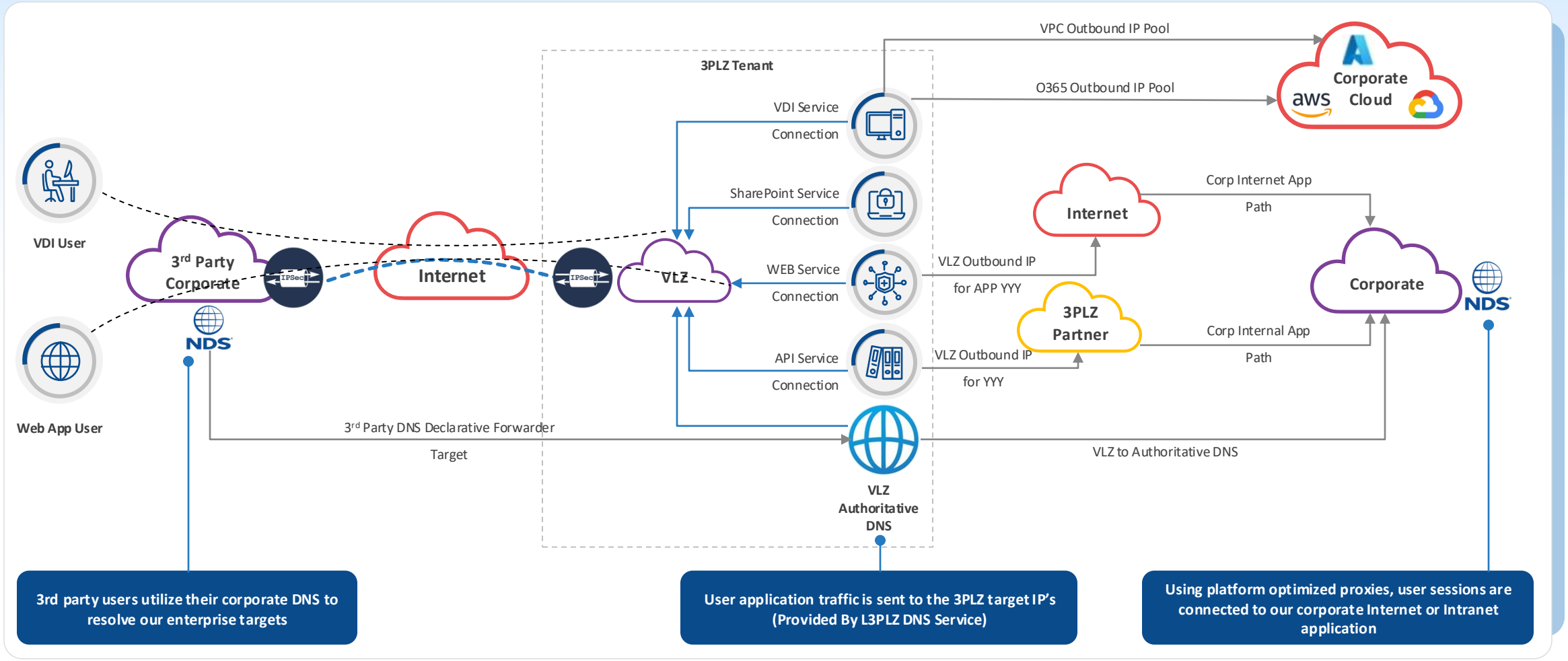
- Each third-party connection **is a dedicated tenant (VLZ)**
- Provides the **enterprise side of the Peer 2 Peer IPSec tunnel**
- **Single HA-FT IPSec Connection** per VLZ
- **VLZ is a no-access zone**, meaning no direct routing to internal networks
- Virtual Landing Zone is **physical, but discrete** (a tenant can have more than one VLZ)

Secure Tenant Model

- A Tenant is **logical** (more than one VLZ can exist in a tenant)
- Each tenant is a **unique identity**, mapped to a business identity
- A VLZ is owned by tenant and is distinctive:
  - Address blocks
  - Tunnel interfaces
  - EBGP ASN
  - DNS Resolver
  - Service Connections
- Availability is built around applications using FQDN reachability

High-Level Architecture View

The architecture connects users seamlessly to our enterprise applications while ensuring robust isolation and compliance.



Next Steps

2025

Apply stakeholder feedback

Update RFP document based on feedback

Send RFP to targeted suppliers
November 21 (or sooner)

Target RFP response by suppliers
December 15

2026

Select top two suppliers, start POC
February 1

Finalize supplier decision and technical design
Q2

Define end-to-end process for 3rd party approval & onboarding
Q2

Will require incremental OPEX in 2026, estimate **TBD** based on POC

Appendix

Application Proxy Solutions Supporting Client-Side mTLS Authentication

Opensource options examined to support mTLS where both client and server authenticate using certificates:

NGINX

- Supports mTLS for reverse proxy and load balancing.
- Can be configured to require client certificates and validate them against a trusted CA.
- Ideal for HTTP(S) applications and API gateways.
- Guide [bastionxp.com]

HAProxy

- Supports SSL termination and mTLS client authentication.
- Useful for high-performance environments and TCP-level proxying.

Envoy Proxy

- Advanced service proxy with native mTLS support.
- Commonly used in service mesh architectures (e.g., Istio).

multi-mtls-proxy (Open Source)

- A Go-based multi-tenant mTLS proxy that handles dynamic certificate management and tenant isolation.
- GitHub Project [github.com]

Apache HTTP Server

- Can be configured for mTLS using mod_ssl.

SSH/SFTP Proxy That Supports Password Authentication And With Password Authentication + SSH Key Forwarding

Use Case	Requirement	Details / Considerations	Product/Capability Needed	Compliance & Regulatory Notes	Vendor Capability (Zscaler, CrowdStrike, Cisco Secure)	Additional Product/Platform Examples
Transparent SSH/SFTP proxy with username/password	Session interception and forwarding	Proxy must transparently forward SSH/SFTP traffic while inspecting payload.	Requires SSH-aware proxy or L4 proxy with DPI (Teleport, HAProxy TCP mode + plugin)	PCI DSS, HIPAA require encrypted file transfers and audit logging.	Zscaler: Partial (SSH passthrough, no deep inspection); CrowdStrike: No proxy; Cisco Secure: Device-level SSH only	Teleport, OpenSSH Bastion, CyberArk, BeyondTrust
Transparent SSH/SFTP proxy with username/password	Credential capture for policy enforcement	Proxy should validate username/password before forwarding.	Integration with PAM or LDAP for credential validation	SOC 2 and ISO 27001 require identity verification and access control.	Zscaler: No credential enforcement; CrowdStrike: Endpoint only; Cisco Secure: Device management	Teleport with PAM, CyberArk, BeyondTrust
Transparent SSH/SFTP proxy with username/password	Traffic inspection	Ability to inspect commands and file operations in SSH/SFTP stream.	Requires SSH protocol parsing (Teleport, SSH Bastion)	GDPR mandates monitoring for sensitive data exfiltration.	Zscaler: No SSH DPI; CrowdStrike: No; Cisco Secure: Limited logging	Teleport with DPI, SSH Bastion, BeyondTrust
Transparent SSH/SFTP proxy with username/password	Audit logging	Log session metadata (user, IP, commands).	Teleport or SSH Bastion with audit logging	HIPAA and PCI DSS require detailed audit trails.	Zscaler: No audit; CrowdStrike: Endpoint logs only; Cisco Secure: Device logs	Teleport with session recording, CyberArk
Transparent SSH/SFTP proxy with username/password but session authenticated via user-specific SSH keys	Key-based authentication enforcement	Proxy must inject user-specific SSH key for backend authentication.	SSH Bastion or Teleport with key management integration	NIST SP 800-63 recommends strong key-based authentication.	Zscaler: No key injection; CrowdStrike: No; Cisco Secure: Supports SSH keys for device mgmt	Teleport + Vault integration, OpenSSH CA
Transparent SSH/SFTP proxy with username/password but session authenticated via user-specific SSH keys	Credential-to-key mapping	Map username/password to corresponding SSH key.	Requires integration with Vault or SSH CA for key issuance	SOC 2 and ISO 27001 require secure key lifecycle management.	Zscaler: No mapping; CrowdStrike: No; Cisco Secure: Limited to device mgmt	HashiCorp Vault, OpenSSH CA, Teleport
Transparent SSH/SFTP proxy with username/password but session authenticated via user-specific SSH keys	Session inspection	Inspect commands and file transfers even when key-based auth is used.	Teleport or advanced SSH proxy with session recording	PCI DSS and HIPAA require monitoring for sensitive data.	Zscaler: No session inspection; CrowdStrike: No; Cisco Secure: Limited	Teleport with session recording, BeyondTrust
Transparent SSH/SFTP proxy with username/password but session authenticated via user-specific SSH keys	Dynamic key rotation	Rotate keys periodically without breaking sessions.	Vault or SSH CA integration with proxy	NIST and FedRAMP mandate cryptographic key rotation policies.	Zscaler: No key rotation; CrowdStrike: No; Cisco Secure: Manual key mgmt	HashiCorp Vault, CyberArk, Teleport

Use case testing conditions and approach

Accepts **password authentication** from the user.

Connects to the target SSH/SFTP host using **SSH keys**.

Other Approaches

Approaches Tested

OpenSSH ProxyJump or ProxyCommand:

Configure the proxy (jump host) to accept password authentication.

ssh -o ProxyCommand="ssh user@proxy-host nc %h %p" target-user@target-host

Agent Forwarding example (or configure the proxy to use its own SSH key for the target host.)

ssh -o ProxyCommand="ssh user@proxy-host nc %h %p" target-user@target-host

sftp -o ProxyCommand="ssh user@proxy-host nc %h %p" target-user@target-host

Proxy Jump example:

ssh -J user@proxy-host target-user@target-host

For SFTP: sftp -o ProxyCommand="ssh user@proxy-host nc %h %p" target-user@target-host

Tools/Utilities

- sshpass can automate password entry for the proxy host.
- Combine with ~/.ssh/config for seamless chaining.

Summary Of Current Vendor/Product Landscape For Virtual Sandboxing

Vendor/Product	Technical Capability Summary	Pros	Cons	Cloud/SaaS Availability	Integration Level	Recommended Use Case
Fortinet FortiSandbox	AI-driven malware detection, multi-layer analysis, custom VM profiles, REST API integration.	Strong integration with Fortinet ecosystem; Flexible deployment (hardware, VM, SaaS); Advanced zero-day detection	Best suited for Fortinet environments; Appliance costs can be high	Yes (FortiCloud)	Fortinet Security Fabric	Enterprise malware analysis and threat detection
Check Point SandBlast	CPU-level threat emulation, OS-level sandboxing, API support, integrated threat prevention.	High accuracy against advanced threats; Tight integration with Check Point security stack	Limited flexibility outside Check Point ecosystem; Premium pricing	Yes	Check Point Harmony Suite	Advanced threat prevention for enterprise security
Cisco Threat Grid	Dynamic & static malware analysis, global threat intelligence, JSON API, custom environment configs.	Rich threat intel feed; Strong API for automation; Flexible deployment options	Requires Cisco ecosystem for full value; Complex initial setup	Yes	Cisco SecureX	Malware analysis with global threat intelligence
FireEye AX Series	Real-time malware analysis, forensic reporting, YARA integration, REST API.	Deep forensic capabilities; Strong reputation in threat analysis	Appliance-heavy model; Higher TCO compared to SaaS-based competitors	Yes	FireEye Helix	Deep forensic malware analysis for SOC teams
VMRay Analyzer	Hypervisor-based sandboxing, evasion resistance, full OS simulation, IOC extraction.	Excellent anti-evasion technology; Detailed behavioral analysis	Higher cost for enterprise scale; Limited non-security use cases	Yes	API Integration	Evasion-resistant malware analysis for security research
ANY.RUN	Interactive sandbox, real-time collaboration, MITRE ATT&CK mapping, script execution.	Highly interactive and user-friendly; SaaS-based, quick onboarding	Limited enterprise integration; Focused on malware analysis only	Yes	API	Interactive malware analysis for SOC and research teams
Joe Sandbox	Deep analysis, YARA rule generation, MITRE ATT&CK mapping, multi-OS support.	Comprehensive analysis depth; Supports multiple OS environments	Requires technical expertise; Pricing can escalate with advanced features	Yes	API	Comprehensive malware analysis across multiple OS environments
AWS Cloud9	Cloud IDE, secure coding, ephemeral environments, IAM integration.	Fully managed SaaS; Tight AWS integration; Ideal for DevOps workflows	AWS lock-in; Limited malware/security sandboxing	Yes	AWS Services	Cloud-based development and secure coding workflows
Azure Sandbox	Secure app testing, CI/CD integration, ephemeral VMs, RBAC.	Seamless with Azure DevOps; Strong RBAC and compliance features	Azure-centric; Limited advanced threat analysis	Yes	Azure DevOps	Secure application testing and CI/CD pipelines
Google Cloud Workbench	AI-powered testing, deployment automation, container isolation.	AI-driven optimization; Strong container support	GCP lock-in; Limited malware analysis capability	Yes	GCP Services	AI-driven testing and containerized app deployments
Red Hat OpenShift Sandbox	Container-based isolation, CI/CD pipelines, multi-tenant support.	Kubernetes-native; Strong multi-tenancy and DevOps integration	Requires container expertise; Not focused on malware analysis	Yes	Kubernetes Ecosystem	Container-native development and multi-tenant isolation
Salesforce Sandbox	CRM-centric app testing, data isolation, API integration.	Ideal for Salesforce developers; Built-in data isolation	Narrow use case (CRM); No malware/security sandboxing	Yes	Salesforce Platform	CRM-centric app development and testing
Thinfinity Workspace	Secure virtual desktops, HTML5 streaming, MFA, RBAC.	Zero Trust architecture; Flexible deployment (cloud/on-prem)	Focused on workspace isolation; Limited malware or dev/test sandboxing	Yes	Zero Trust Architecture	Secure virtual desktops and remote workspace isolation
Kasm Workspaces	Web-native DaaS, container isolation, ephemeral sessions, policy enforcement.	Strong isolation via containers; API-driven automation	Requires container knowledge; Limited advanced threat analysis	Yes	API	Web-native DaaS with container isolation for secure app hosting
Cameyo	Secure app delivery, no VPN required, HTML5 streaming, MFA.	Lightweight and easy to deploy; Strong security posture	Focused on app delivery; Not suitable for malware or deep dev/test environments	Yes	API	Secure app delivery without VPN for remote work scenarios

Session Isolation And Attack Sandboxing Capabilities Across 3 Vendors

Capability	Cloudflare	Zscaler	Cisco Secure Cloud
Session Isolation for Suspicious Users	Browser Isolation: Redirect risky sessions to remote browser sandbox; prevents exploit execution on real app.	Cloud Browser Isolation (CBI): Streams content in isolated container while sandbox verdict is pending.	Remote Browser Isolation (RBI): Executes high-risk sessions in virtual browser; integrated with SSE stack.
Sandbox for Exploit Attempts	Sandbox SDK: Runs user-specific sandboxes in isolated VMs; supports dynamic code execution for exploit analysis.	Cloud Sandbox: Detonates suspicious files and URLs in full OS emulation; observes exploit behaviors.	Threat Grid (Secure Malware Analytics): Detonates files and scripts in isolated environments for exploit detection.
Forensic Data Collection	Logs sandbox activity, command execution, and network calls; supports tokenized preview URLs for controlled access.	Generates detailed behavioral reports, IOCs, and integrates with ThreatQuotient for forensic workflows.	Secure Workload Forensics: Captures MITRE ATT&CK-based forensic events, process lineage, and anomaly detection.
Dynamic Risk-Based Triggering	Policies via Cloudflare Access and Gateway to enforce isolation for suspicious identity/device posture.	ZIA/ZPA Zero Trust policies trigger sandboxing and isolation based on user/device risk score.	Cisco Secure Access enforces sandboxing and forensic monitoring based on contextual risk signals.
Exploit Behavior Monitoring	Observes system calls and network activity inside sandbox; supports adversarial simulation for exploit detection.	Detects privilege escalation, C2 communication, encryption attempts during sandbox detonation.	Monitors lateral movement, persistence techniques, and network anomalies during sandbox execution.
Integration with Zero Trust	Integrated with Cloudflare Zero Trust stack for identity-based isolation and logging.	Works with Zscaler Zero Trust Exchange for inline sandboxing and isolation.	Cisco SSE combines sandboxing, RBI, ZTNA, and forensic analytics under unified policy engine.

Key Virtual Sandboxing Technical Capabilities

Capability	Data Collected During Attack	Ideal Deployment Scenarios	Integration Complexity	Decryption of Data Stream	Rate Limiting Implementation	Traffic Pattern Matching	Data Validation
Session Isolation for Suspicious Users	Session logs, user actions, network requests	High-risk user sessions, unmanaged devices	Low (policy-based)	TLS termination at edge	Supported via Gateway policies	Behavioral anomaly detection	Basic input sanitization and HTTP validation
Sandbox for Exploit Attempts	System calls, exploit payloads, memory dumps	Dynamic exploit analysis, malware detonation	Medium (requires sandbox orchestration)	Full decryption for sandbox inspection	Rate limiting via API Gateway or Workers	Pattern matching for exploit signatures	Deep validation of code execution and file integrity
Forensic Data Collection	IOC generation, process lineage, network traces	Incident response and threat hunting workflows	Medium (integration with SIEM/EDR)	Decryption for forensic packet capture	Configurable throttling for forensic pipelines	Traffic fingerprinting for anomaly detection	Validation of forensic data integrity
Dynamic Risk-Based Triggering	Risk scores, identity posture, device compliance	Adaptive Zero Trust enforcement	Low (policy-driven)	Decryption for identity verification	Rate limiting based on risk score	Pattern matching for suspicious behavior	Validation of identity and session tokens
Exploit Behavior Monitoring	Privilege escalation attempts, C2 traffic logs	Advanced persistent threat detection	High (requires behavioral analytics engine)	Decryption for deep packet inspection	Dynamic throttling during exploit simulation	Behavioral pattern recognition	Validation of exploit signatures and payload integrity
Integration with Zero Trust	Access logs, policy enforcement actions	Enterprise-wide Zero Trust deployments	Medium (requires orchestration across services)	Decryption for secure access enforcement	Rate limiting for policy enforcement endpoints	Traffic pattern matching for compliance	Validation of Zero Trust policy execution

Service Patterns and Service Connections

Service Pattern

Users provide connectivity using application specific proxy configuration templates:

- SFTP/SSH proxy, VDI proxy, Web App proxy, Web API proxy, SaaS proxy

Each service has a set of templates that are optimized for different use cases (e.g., Internet Reachable Target, SFTP using blind key authentication).

Application owners define Service Profiles via the portal:

- Choose service type (e.g., Enterprise Web App, Internet SaaS)
- Provide FQDNs, ports, and header/path requirements
- Select controls (IP restrictions, auth requirements, rate limiting)

Once the use case pattern is selected, to define a service connection all that is needed is the user to define the corporate FQDN application target.

Service Connections

- Platform translates profiles into proxy configs, ACLs, and DNS rules
- FQDN/IP defined application connections that are reachable by 3rd party users over the IPSec tunnel between the 3rd party network and the 3PLZ VLZ
- 3Rd party users use there corporate DNS to resolve applications. VLZ DNS service rewrites DNS lookup requests to reflect the VLZ proxy address
- Reachability to application service connections can be tied to successful corporate DNS resolution and if needed remote hop verification tied to BGP announcements

Sample Portal Form – Secure File Transfer Service-Pattern (SFTP Service-Connection VLZ Proxy)

Example fields collected in the portal

- Business context: application name, owner, environment (dev/test/prod)
- Third-party details: organization name, contact info, support hours
- Network endpoints: remote public IP/FQDN, device type/platform
- Service specifics: SFTP hostnames, ports, internal target FQDNs
- Data sensitivity & compliance tags (PHI/PII, region, retention)
- Operational parameters: expected volume, schedules/maintenance windows

Validations & policy checks

- Required fields, allowed port ranges, approved cipher suites
- Automatic mapping to a standard SFTP pattern and VLZ tenant
- Routing to appropriate approvers and risk/compliance reviewers

Sample Portal Form – Web API (Enterprise API via VLZ Proxy)

Example fields collected in the portal

- Business context: application name, owner, environment (dev/test/uat/prod)
- API details: API name, internal FQDN, base path (e.g., /v1/payments)
- Third-party details: organization, contact, support window, region(s)
- Network endpoints: remote public IP(s)/FQDN(s), expected source CIDRs
- Auth & identity: auth type (mTLS, OAuth2/JWT, API key), required claims/scopes
- Data classification: sensitivity (PHI/PII), residency, retention, logging level
- Traffic profile: expected QPS, burst limits, concurrency, allowed methods

Validations & policy checks

- Required fields, allowed methods and ports, approved cipher suites
- API classification drives mandatory controls (e.g., mTLS for high sensitivity)
- Automatic mapping to a standard Web API pattern and VLZ tenant

Web API Service Behavior & Controls

Service proxy behavior

- Exposes a tenant-specific API endpoint (e.g., https://<tenant-id>.api.3plz.example.com)
- Forwards to internal API FQDN/base path using mTLS where supported
- Normalizes paths, headers, and TLS parameters based on the Service Profile

Security controls

- Enforces allowed HTTP methods, paths, and rate limits per profile
- Supports mTLS, JWT/OAuth2 validation, or API key verification
- Optional IP/CIDR restrictions per third party / tenant
- Request/response size limits and protocol sanity checks

Observability & governance

- Per-tenant logs with correlation IDs, status codes, latency, and auth outcome
- Metrics export (QPS, error rates, throttling events) to observability stack
- Logs tagged with tenant, vendor, application ID, and data classification

From Form → IPsec Configuration & B2B Transport

Each approved request generates a remote IPsec configuration bundle

- IKEv2 parameters, proposals, and crypto settings
- Tunnel interface addresses and BGP configuration (if applicable)

Config bundles must support major firewall/router platforms

Local side terminates on location-agnostic tunnel anchors

All tunnel creation driven by templates and portal data, not manual builds

Identity, DNS, and Observability

Identity & Authorization: RBAC roles in the portal and APIs

Tenant-linked identities influence policy decisions and approvals

DNS as a control plane: per-tenant views, split-horizon, conditional forwarding

Per-tenant logs for DNS, proxy traffic, and tunnel/BGP state

Logs and metrics must integrate with enterprise SIEM/observability tools

Automation & Scale Requirements

All portal operations available via documented APIs

Support for IaC tools (e.g., Terraform, Ansible) and CI/CD integration

Target scale: ~1000 tenants, multiple patterns per tenant

High availability for control and data planes

Standards-compliant crypto and minimal-disruption upgrades

Shared Responsibilities: Business vs Vendor Platform

Business responsibilities

- Provide an application governance mechanism with Universal Application IDs
- Provide a Universal Identifier for third-party partners (3pBID)
- Supply Universal IDs and 3pBIDs to drive tenant creation and site-to-site VPN setup
- Operate PKI processes to generate and deliver mTLS certificates for Web/WebAPI tenants
- Operate enterprise IdPs (SAML/OAuth2/OIDC) for Virtual Tenant portal SSO and RBAC

Vendor platform responsibilities

- Treat Universal IDs and 3pBIDs as first-class attributes in tenant and config APIs
- Automate VPN and service proxy creation from portal data and identifiers
- Integrate with enterprise PKI for certificate-based auth and renewal workflows
- Integrate with enterprise IdPs for authentication and authorization to the portal

What We Expect in Vendor Responses

Describe end-to-end portal-driven workflows

- From app owner request → IPsec config → service proxy live

Show how your platform implements the required service patterns

Demonstrate self-service forms that generate

- Remote IPsec bundles, VLZ/tenant config, service proxy rules

Provide reference architectures, runbooks, and automation artifacts

Map capabilities to security, compliance, scale, and shared responsibility model


CVS Health®