

Labyrinth Deception Platform

Customer Presentation

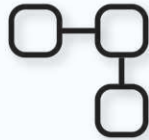
Choose innovation. Choose proactive defence.
Choose Deception Technology



Labyrinth Security Solutions, 2025

Cybersecurity challenge

Reactive approach to
threat detection



False positive alarms



Difficult in usage



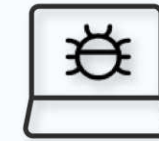
Information overload



More time to
detect and react

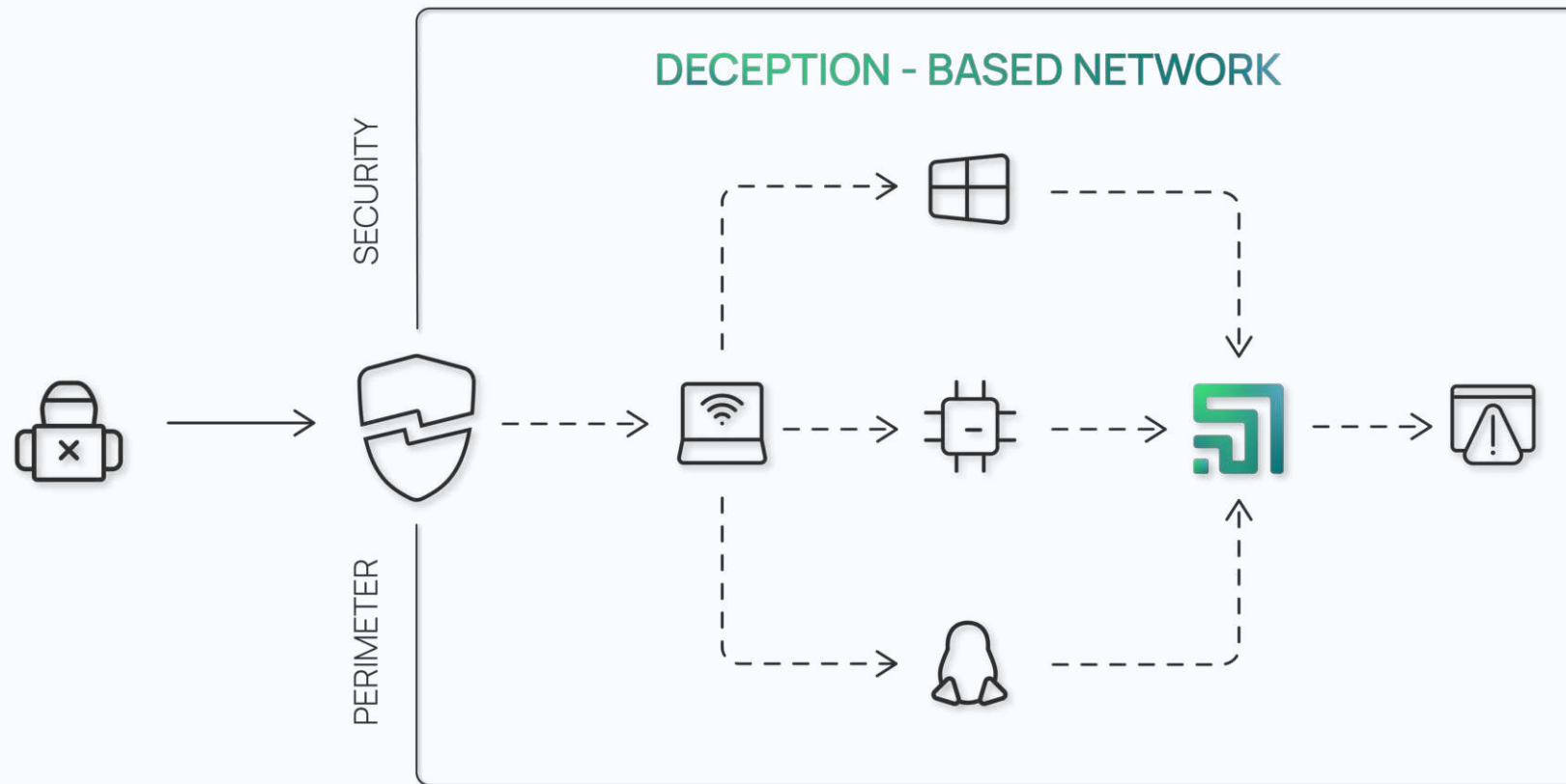


Breaches



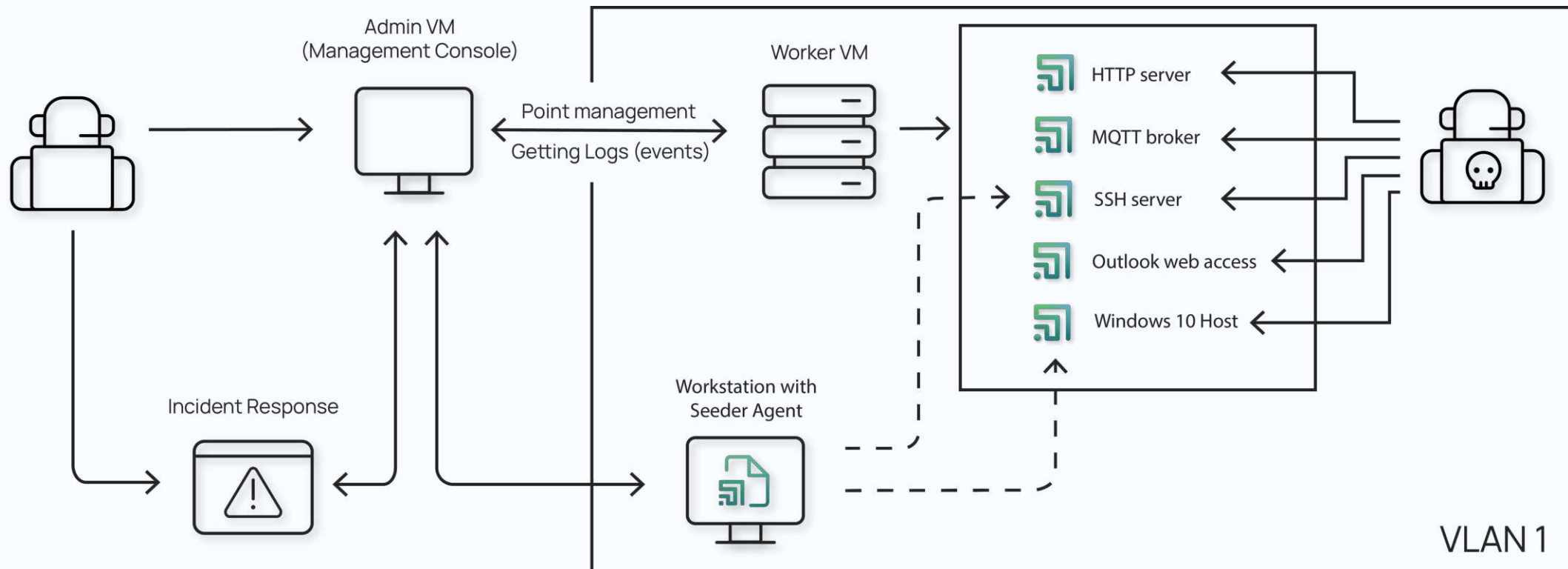
Deception-based threat detection

The Labyrinth Deception Platform is changing the cybersecurity paradigm by taking a proactive approach to threat detection.



Labyrinth Deception Platform

The platform creates vulnerable IT services and applications, increasing the attack surface and disorienting attackers. The Labyrinth provokes attackers to act, detects and tracks all their activities, and isolates them from the actual IT network.

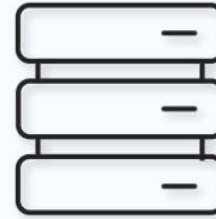


Business values



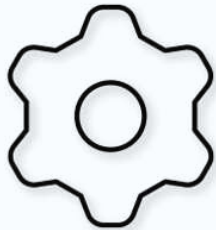
Stops sophisticated threats

Detects targeted and advanced attacks without requiring prior knowledge of the threat's form, type, or behavior.



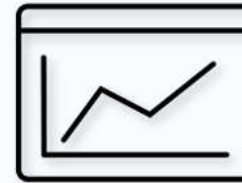
Zero impact on performance

No negative impact on the performance of network devices, hosts, servers, or applications behavior.



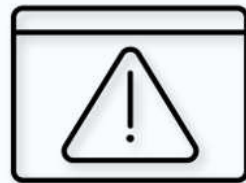
Simple implementation

Quick and easy deployment with no system conflicts and minimal maintenance: no databases, signatures, or rules to configure and update.



Operation costs reduction of by 30%*

Doesn't collect tons of data, doesn't generate false positive alerts, doesn't require special skills to operate.



Incident response automation

Speeds up incident response by reducing the average time to detection and response (MTTD, MTTR) by up to 12** times.

* https://www.enterprisemanagement.com/news/press_release.php?p_id=2659

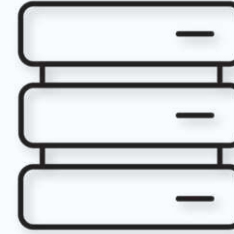
** <https://www.bloomberg.com/press-releases/2020-09-14/cyber-deception-reduces-data-breach-costs-by-over-51-and-soc-inefficiencies-by-32>

Labyrinth's components



Admin VM (Management Console)

All information collected at the Points is forwarded to the Management Console for incident analysis and response.



Worker VM

The Worker VM is the host that hosts all the Points in Labyrinth. It can operate in multiple VLANs simultaneously.



Point

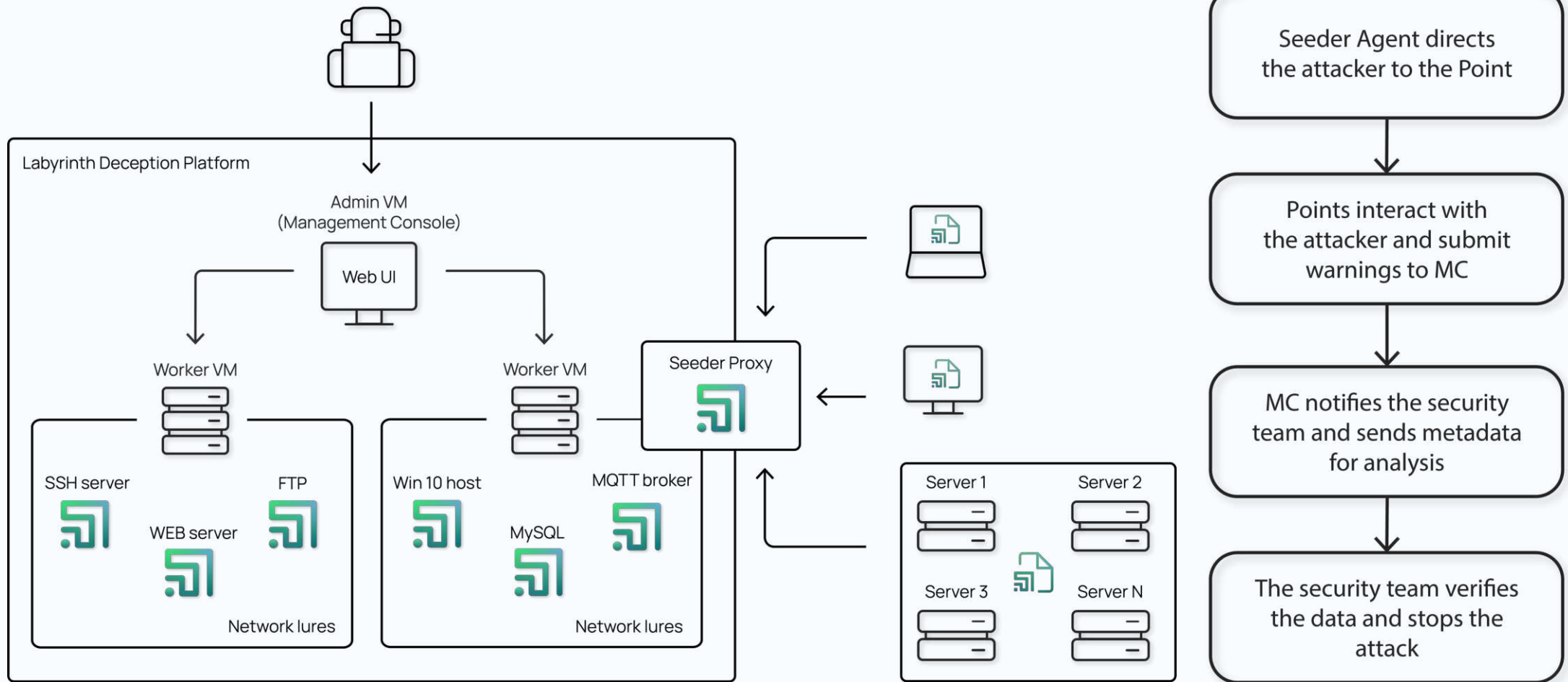
Points simulate applications and services in a real-world IT environment and interact with attackers, keeping them inside the Labyrinth.



Host with Seeder Agent

Agents are deployed on real hosts and distribute attractive artifacts to them. The artifacts used by attackers direct them to Points.

Solution architecture



Points

POINT TYPES POINT TYPE BASES

Id ↑	Name	Default	Tags	Description
1c	1C8.1	✓	1c.web	1C: Предприятие Web login page
ab_ethp	Allen Bradley Ethernet Processor...	✓	web.scada.ot	Allen Bradley Ethernet Processor SLC-500 (1747-L552/C)
ab_plc	Allen Bradley PLC	✓	web.scada.ot	Allen Bradley PLC CompactLogix 5069-L32C
askod	АСКОД WEB	✓	askod.web	АСКОД WEB Login page imitation (Ukraine)
clientes	Workstation	✓	workstation.client_desktop	Workstation network activity imitation and MI
dns_bind	DNS server with AXFR	✓	dns	DNS server with AXFR enabled (zone transf
dns_bind_wo_axfr	DNS server (AXFR disabled)	✓	dns	DNS server with AXFR disabled (zone transf

Points provide services tailored to various sectors, ranging from Basic IT to OT and IoT.

Each decoy can be easily customized through user-friendly YAML configuration, allowing to adjust solution to your specific needs.

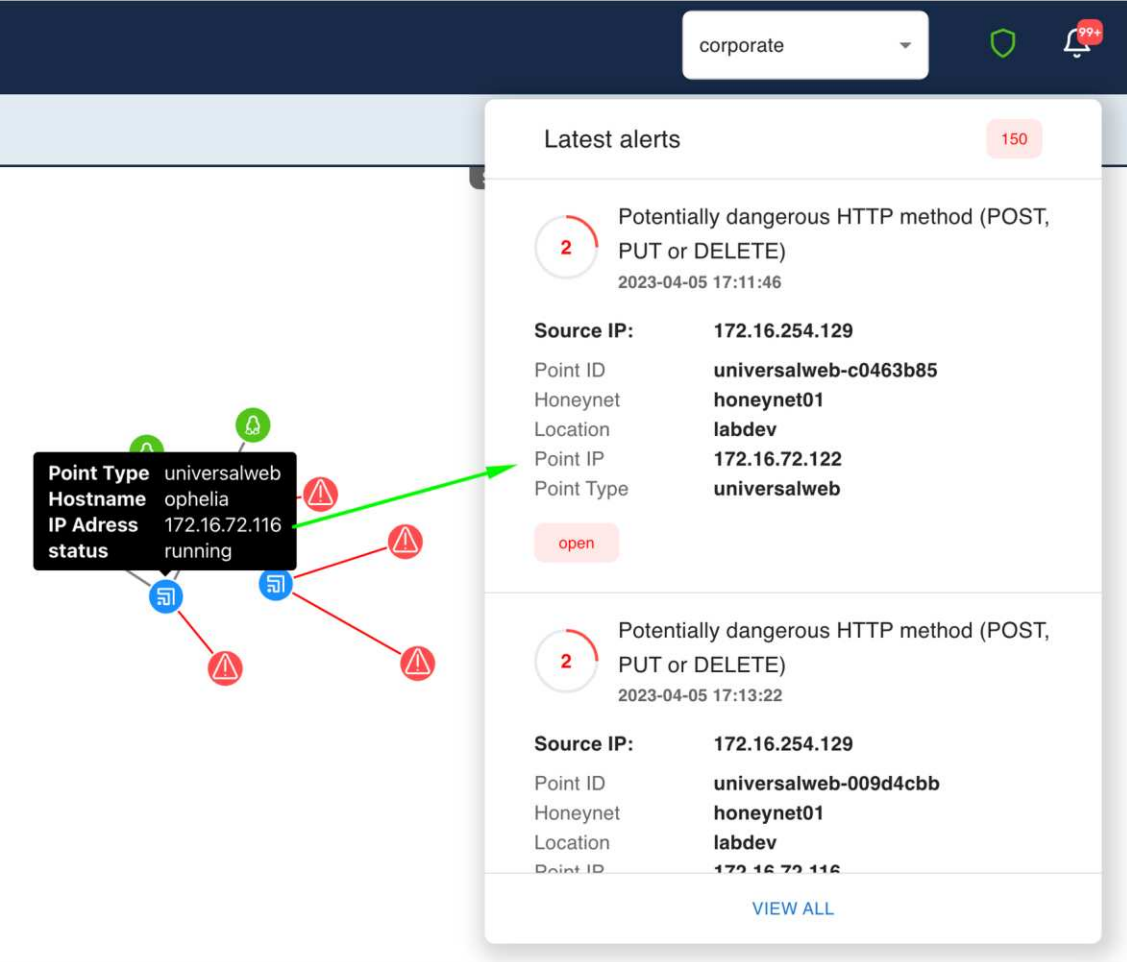
Point config

```
1  ## Option: hostname
2  ## Required: no
3  ## Description: hostname value is short domain name or Fully Qualified Domain Name of the
4  ##                Point host.
5  ##                This option may be omitted. In this case hostname value will be generate for
6  ##                each instance of a Point:
7  ##                1. from hostnames wordlist which is specified in Point Type configuration
8  ##                2. from hostnames wordlist which is specified in Honeynet configuration.
9  #
10 # hostname: my.host.name
11 #
12 ## Option: fake_ports
13 ## Required: no
14 ## Description: fake_ports are TCP and UDP ports which will be visible to network scanners as
15 ##                filtered ports.
16 ##                Main goal of fake ports is simulation of services which are binds to Point's
17 ##                ports but are filtered by firewall.
18 ##                Actually there is no any service which is listening on fake ports.
19 ##                fake_ports is a list of objects which have tcp and udp properties or just tcp
20 ##                or udp.
21 ##                During generate process for each instance of Point which uses current
22 ##                configuration will be randomly chosen one of fake ports groups.
23 #
24 # fake_ports:
```


Universal Web Point

Attackers most often use web application vulnerabilities to hack into corporate networks.

Labyrinth has implemented a unique technology that provides additional protection for the most used targets by hackers - **web applications and services**.



The screenshot displays the Labyrinth Deception Platform interface. At the top, there is a dark blue header with a dropdown menu set to 'corporate', a shield icon, and a notification bell with '99+'. Below the header, a network diagram shows several nodes connected by lines. A tooltip is visible over one node, displaying the following information:

Point Type	universalweb
Hostname	ophelia
IP Address	172.16.72.116
status	running

To the right of the network diagram, a 'Latest alerts' panel is shown, containing two alert entries. Each entry includes a red circular icon with the number '2', the alert text 'Potentially dangerous HTTP method (POST, PUT or DELETE)', the timestamp '2023-04-05 17:11:46' (or '17:13:22'), and a list of details:

- Source IP:** 172.16.254.129
- Point ID:** universalweb-c0463b85 (or universalweb-009d4cbb)
- Honeynet:** honeynet01
- Location:** labdev
- Point IP:** 172.16.72.122 (or 172.16.72.116)
- Point Type:** universalweb

Each alert entry has an 'open' button below it. At the bottom of the alerts panel, there is a 'VIEW ALL' link.

Universal Web Point

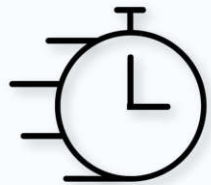
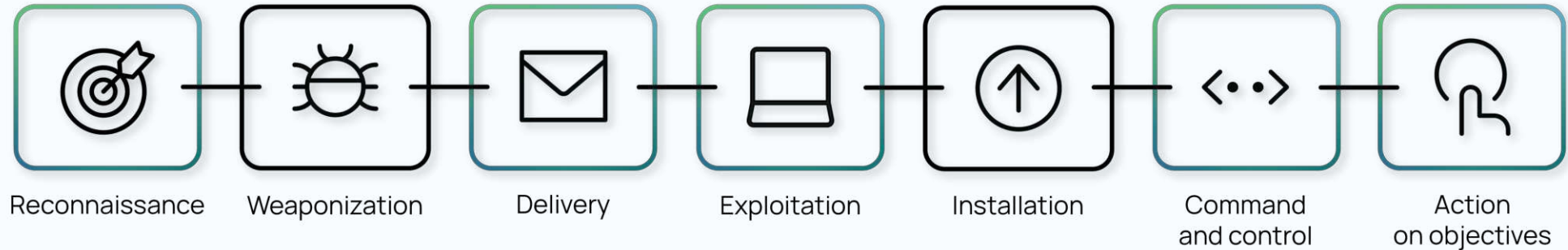
The image displays two side-by-side browser windows showing a Cisco Switch login page. The left window is from IP 192.168.200.20 and the right is from 192.168.200.32. Both show the same login form with fields for Username, Password, and Language, and buttons for Log In and Secure Browsing (HTTPS). Below the browser windows are two Network Inspector panels showing request logs. Red boxes highlight the 'Domain' column in both logs, which shows requests to 192.168.200.20 on the left and 192.168.200.32 on the right.

Status	Method	Domain	File	Initiator	Type	Transferred	Size
200	GET	192.168.200.20	button.gif	img	gif	6.47 KB	6.26 KB
200	GET	192.168.200.20	favicon.gif	FaviconLoader.jsm:1...	gif	1.33 KB	1.12 KB
200	GET	192.168.200.20	logo_cis.gif	log_off_page.htm:5...	gif	891 B	678 B
200	GET	192.168.200.20	pageBackground.jpg	log_off_page.htm:5...	jpeg	14.85 KB	14.64 KB
200	GET	192.168.200.20	Status_information_icon.png	log_off_page.htm:5...	png	2.29 KB	2.08 KB
200	GET	192.168.200.20	ContextMessageArrow_DownT.gif	log_off_page.htm:5...	gif	1.03 KB	839 B
200	GET	192.168.200.20	login_progress.gif	log_off_page.htm:5...	gif	886 B	673 B
200	GET	192.168.200.20	topLeft.gif	log_off_page.htm:5...	gif	1 KB	816 B
200	GET	192.168.200.20	topRight.gif	log_off_page.htm:5...	gif	1 KB	816 B
200	GET	192.168.200.20	bottomLeft.gif	log_off_page.htm:5...	gif	1 KB	816 B
200	GET	192.168.200.20	bottomRight.gif	log_off_page.htm:5...	gif	1 KB	816 B
200	GET	192.168.200.20	bar.gif	log_off_page.htm:5...	gif	0.99 KB	801 B

Status	Method	Domain	File	Initiator	Type	Transferred	Size
200	GET	192.168.200.32	button.gif	img	gif	6.47 KB	6.47 KB
200	GET	192.168.200.32	favicon.gif	FaviconLoader.jsm:1...	gif	1.33 KB	1.33 KB
200	GET	192.168.200.32	logo_cis.gif	log_off_page.htm:5...	gif	891 B	891 B
200	GET	192.168.200.32	pageBackground.jpg	log_off_page.htm:5...	jpeg	14.85 KB	14.85 KB
200	GET	192.168.200.32	Status_information_icon.png	log_off_page.htm:5...	png	2.29 KB	2.29 KB
200	GET	192.168.200.32	ContextMessageArrow_DownT.gif	log_off_page.htm:5...	gif	1.03 KB	1.03 KB
200	GET	192.168.200.32	login_progress.gif	log_off_page.htm:5...	gif	886 B	886 B
200	GET	192.168.200.32	topLeft.gif	log_off_page.htm:5...	gif	1 KB	1 KB
200	GET	192.168.200.32	topRight.gif	log_off_page.htm:5...	gif	1 KB	1 KB
200	GET	192.168.200.32	bottomLeft.gif	log_off_page.htm:5...	gif	1 KB	1 KB
200	GET	192.168.200.32	bottomRight.gif	log_off_page.htm:5...	gif	1 KB	1 KB
200	GET	192.168.200.32	bar.gif	log_off_page.htm:5...	gif	0.99 KB	0.99 KB

Labyrinth generates sophisticated emulations of existing web resources, known as **Universal Web Points (UWP)**. These emulations are further enhanced by embedding additional vulnerabilities, making them more enticing targets for attackers.

Use cases



- Early detection of network threats
- Proactive protection
- Targeted attack detection
- Reduced Dwell Time



- Man-in-the-Middle detection
- Lateral Movement identification
- Rapid response to incidents
- Incident investigation

Use case scenario: stolen credentials

```
~ % ssh test3.test2@172.16.132.28
The authenticity of host '172.16.132.28 (172.16.132.28)' can't be established.
ED25519 key fingerprint is SHA256:XEYAhSySo8BfIu8k/5l+iXZ+Wr6Itfynjptz+KEbnc.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '172.16.132.28' (ED25519) to the list of known hosts.
test3.test2@172.16.132.28's password:
test3.test2@tethys:~$ whoami
test3.test2
test3.test2@tethys:~$
```

<input type="checkbox"/>	Severity	Status	Timestamp	Point ID	Attacker IP	Alert Reason	
<input type="checkbox"/>	L	open	2024-06-02 20:52:30	sshd-26e9adf2	172.16.254.4	Connection to sshd port detected	^

DETAILS	EVENTS	ACTIVITY(0)
2024-06-02 20:53:22	Hostname: - Message: CMD: whoami	
2024-06-02 20:52:37	Hostname: -	
2024-06-02 20:52:37	Hostname: - Message: Terminal Size: 176 50	
2024-06-02 20:52:37	Hostname: - Username: test3.test2 Message: login attempt [test3.test2/15061988] succeeded	
2024-06-02 20:52:37	Hostname: - Name: LC_CTYPE Message: request_env: LC_CTYPE=UTF-8	
2024-06-02 20:52:30	Hostname: - Message: SSH client hassh fingerprint: aae6b9604f6f3356543709a376d7f657	

Use case scenario: network scanning

□ L open 2024-06-02 21:37:56 win_generic-6cf15eea 172.16.254.4 Port scan detected (TCP SYN e.g. nmap -...

[DETAILS](#) [EVENTS](#) [ACTIVITY\(0\)](#)

2024-06-02 21:37:56

Alert ID	767944f8-1d05-49eb-ba17-6d2c254398b1
Alert Reason	Port scan detected
Destination IP	172.16.132.30
MITRE	
Technique	T1595
Tactic	TA0043

```
~ % nmap 172.16.132.30
Starting Nmap 7.94 ( https://nmap.org ) at 2024-06-02 21:38 EEST
Nmap scan report for 172.16.132.30
Host is up (0.031s latency).
Not shown: 992 closed tcp ports (conn-refused)
PORT      STATE SERVICE
135/tcp   open  msrpc
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
5357/tcp  open  wsdapi
49152/tcp filtered unknown
49153/tcp filtered unknown
49154/tcp filtered unknown
49156/tcp filtered unknown

Nmap done: 1 IP address (1 host up) scanned in 5.93 seconds
```

Use case scenario: web scanning

The screenshot displays the Labyrinth Deception Platform interface. At the top, a search bar contains the IP address "192.168.200.201". Below the search bar, a modal window is open, displaying the following information:

- Web scanner has been detected**
- Point Info**
 - Point ID: vmware_esx-b3aa40df
 - Point IP: 192.168.200.45
 - Point Type: vmware_esx
- Attacker Info**
 - Source IP: 192.168.200.201
 - Reason: Web scanner has been detected
 - Alert Score: 1
 - Risk Score: 2010
- IR Info**
 - Status: open
 - IR Link: N/A (Case not created yet)
- Timestamp:** 13.04.2021 18:37:05

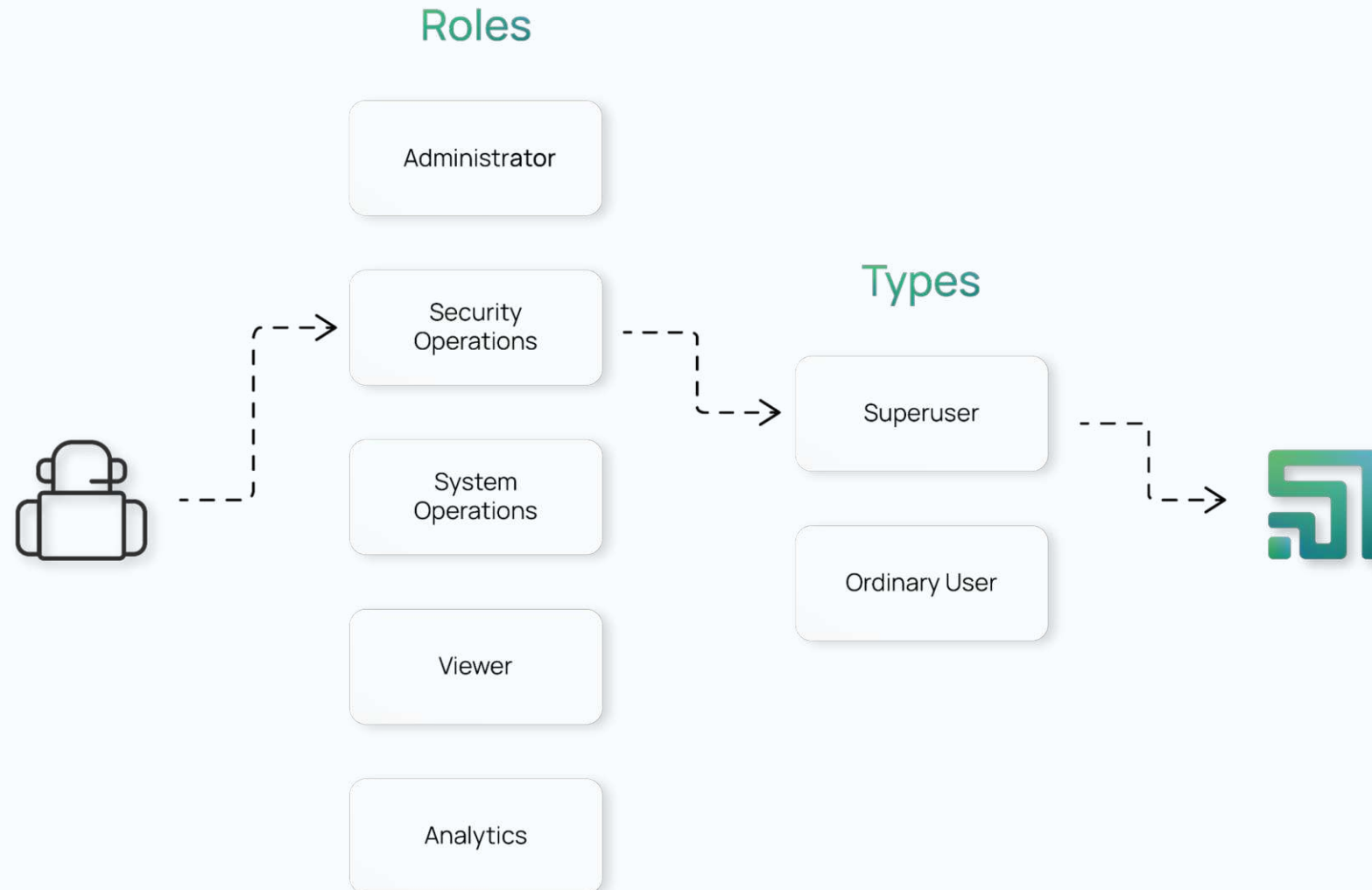
The background shows a network map with various nodes and connections. A red arrow points from the search bar to the "Attacker Info" section of the modal. The interface includes a "Map" tab, a "Controls" panel at the bottom, and a sidebar with "Minimap", "Actions", and "Legend" buttons.

Multitenancy

The screenshot displays the Labyrinth web interface. The top navigation bar includes the Labyrinth logo, a dropdown menu set to 'department1', and icons for a shield, notifications, user profile, and a share icon. A dark sidebar on the left contains a menu with items: Dashboard, Honeynets, Points, Seeder Agents, Map, Alerts, Audit Log, Nodes, Multitenancy (highlighted), Settings, and License. The main content area is titled 'Tenant list' and shows 'Tenant license used: 6 available: 10' with an 'ADD' button. Below this is a table with columns for Name, Honeynet (VLAN) (used / reserved), Points (used / reserved), and Actions. The table lists seven tenants: demo, client, main_office, department1, advanced_cases, and additional_tenant, each with edit and delete icons.

Name	Honeynet (VLAN) (used / reserved)	Points (used / reserved)	Actions
demo	1/2	4/50	edit delete
client	2/4	27/90	edit delete
main_office	1/3	10/90	edit delete
department1	1/3	12/90	edit delete
advanced_cases	2/5	19/70	edit delete
additional_tenant	2/10	36/70	edit delete

RBAC: system users



Integrations



State	Name	Edit
	CrowdStrike	Edit
	FortiGate	Edit
	Microsoft Teams Notifications	Edit
	IBM QRadar	Edit
	Slack Notification	Edit
	SMTP Notification	Edit
	Splunk	Edit
	SIEM Integration (Syslog forwarder)	Edit
	TheHive	Edit
	Trellix	Edit
	Webhook	Edit

API

Overview

Authentication

Content-Type

Timezone

Resource Group

Get status

Get License Info

List All Tenants

List All Nodes

Audit logs

Autocomplete

List existing logs

List All Honeynets

Points

List All Points

Get Point Details

Manage Point

Delete Point

Alerts

List All Alerts

Get Alert Details

Manage Alert

Download traffic dump

List All Seeders

List Seeders Tasks

<https://your-provider-host.com/api/v1>

Labyrinth API v.1

The Labyrinth API provides a way to manage Labyrinth resources. The API is compliant to all REST standards: resource-oriented URLs, returns JSON-encoded responses, uses standard HTTP response codes and verbs, authenticates and communicates through secure HTTPS connections.

AUTHENTICATION

Each request must be authenticated with private token. The API v.1 uses Bearer authentication scheme. Building authorization header example: `Authorization: Bearer <token>`

CONTENT-TYPE

Needs to be "application/json" for POST and PUT, but "" for GET and DELETE.

TIMEZONE

The times returned are in UTC.

Resource Group

API STATUS

GET /status Get status

Example URI

GET `https://your-provider-host.com/api/v1/status`

Response 200 Show

Response 401 Show

Response 429 Show

LICENSE

GET /license Get License Info

Example URI

GET `https://your-provider-host.com/api/v1/license`

Response 200 Show

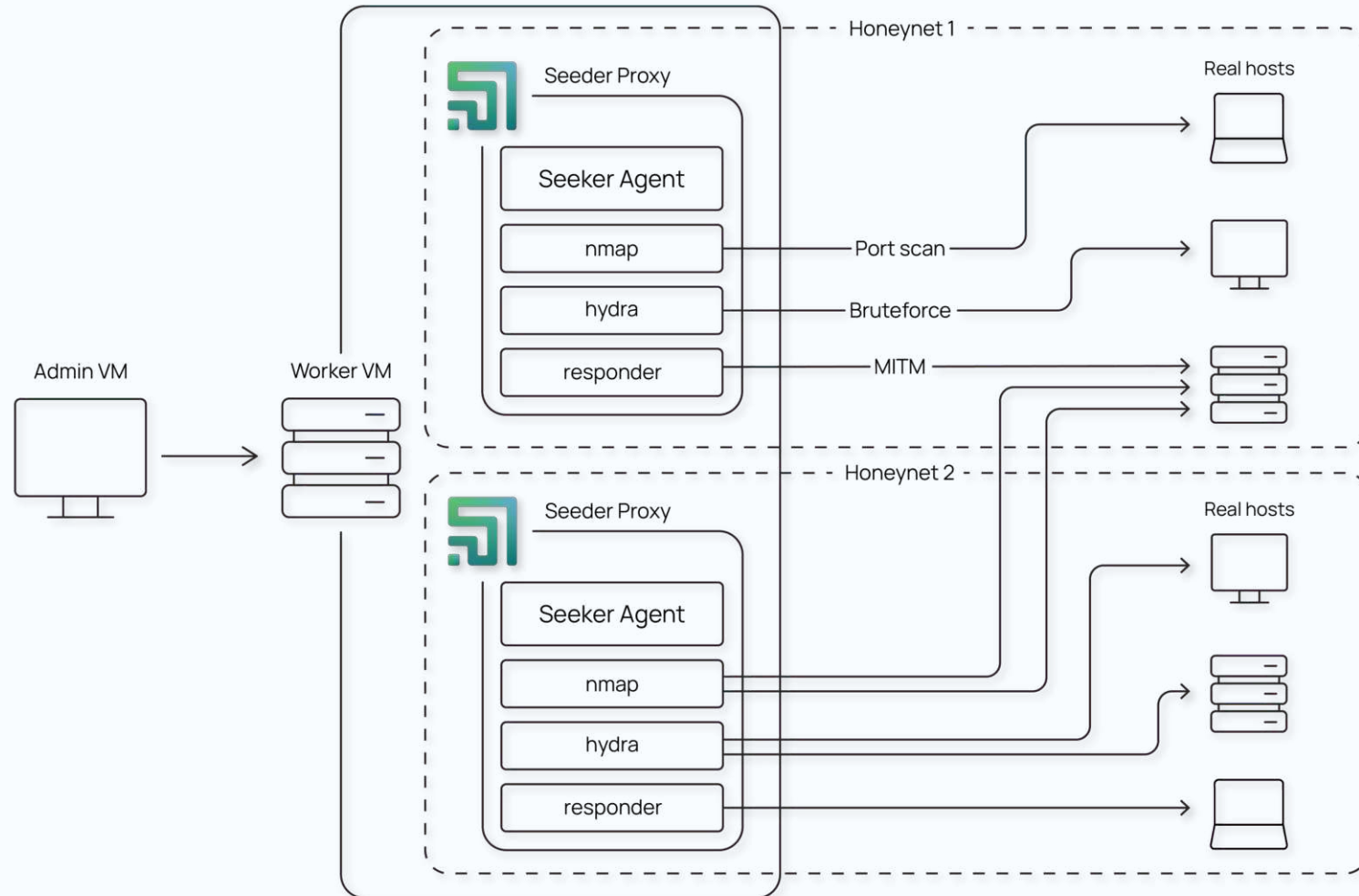
Response 401 Show

Response 429 Show

TENANTS

[Back to top](#)

Seeker



LABYRINTH

Labyrinth is a team of experienced cybersecurity engineers and penetration testers, which specializes in the development of solutions for early cyber threat detection and prevention.

Follow us on:



Labyrinth Development



Labyrinth Deception Platform



<https://labyrinth.tech>



info@labyrinth.tech

