



SECURITYBOAT
Frontline Of Your Business

PRIVILEGE ESCALATION HANDBOOK

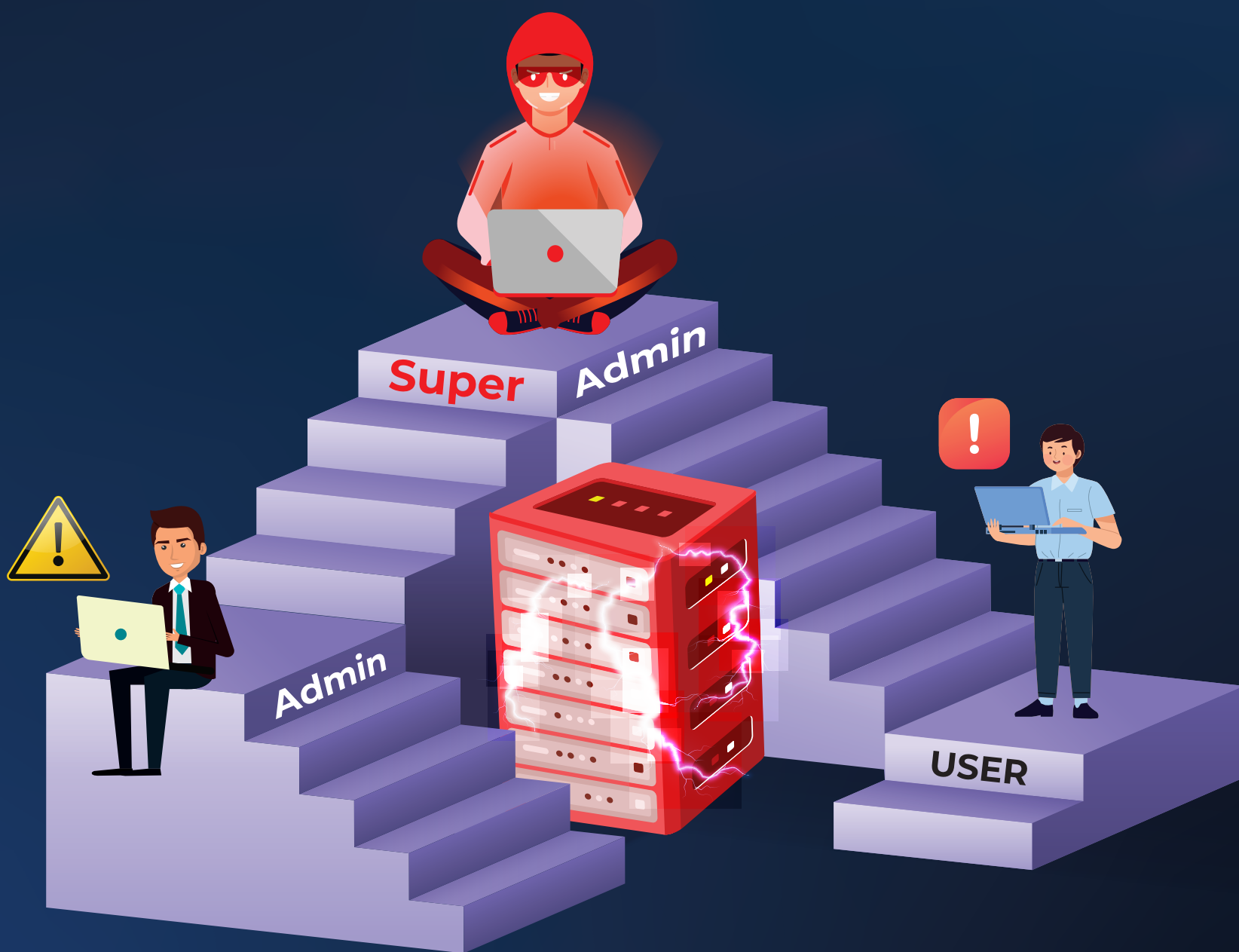




Table of Contents

1. What is Authentication and Authorization?	01
2. What is Privilege escalation	01
3. Types of Privilege Escalation	01
3.1 Linux Command Injection	01
3.2 Windows Command Injection	01
4. Vulnerability that leads to privilege escalation	02
4.1 Cross-Site Scripting (XSS)	02
4.2 SQL Injection	02
4.3 Insecure Direct Object References (IDOR)	02
4.4 Broken Access Control	03
4.5 Local File Inclusion (LFI)	03
4.6 Server-Side Request Forgery (SSRF)	03
4.7 XML External Entity (XXE)	03
5. How to Detect Privilege Escalation	04
5.1 Log Analysis	04
5.2 Intrusion Detection Systems (IDS)	04
5.3 Security Information and Event Management (SIEM)	04
5.4 User Behavior Analytics (UBA)	04
6. Impact of Privilege escalation	05
6.1 Unauthorized Access	05
6.2 Data Theft and Manipulation	05
6.3 Compromise of User Accounts	05
6.4 Service Disruption	05
6.5 Reputational Damage	05
6.6 Legal and Regulatory Consequences	05
7. Remediation	06
7.1 Implement Secure Authentication	06
7.2 Apply Principle of Least Privilege (PoLP)	06
7.3 Secure Session Management	06
7.4 Employ Defense-in-Depth Strategies	06
7.5 Monitor and Control API Access	06
8. Additional Resources and References	07
9.QR Code	08





1. What is Authentication and Authorization?

Authentication is the process of verifying the identity of a user or system. It ensures that the entity requesting access is indeed who they claim to be. Common methods include passwords, biometrics, and multi-factor authentication.

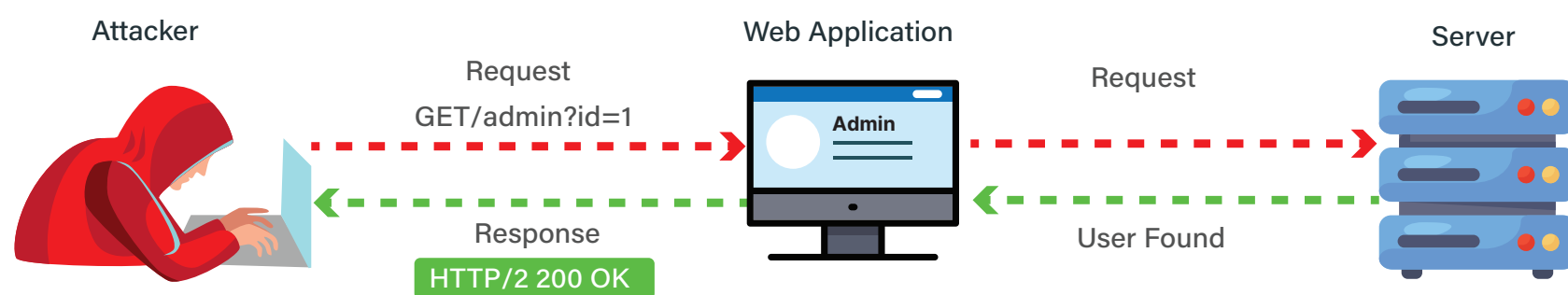
Authorization determines what resources an authenticated user or system can access. It involves granting permissions and access levels based on the entity's identity. This process ensures that users can only access data and perform actions that they are permitted to.

2. What is Privilege Escalation

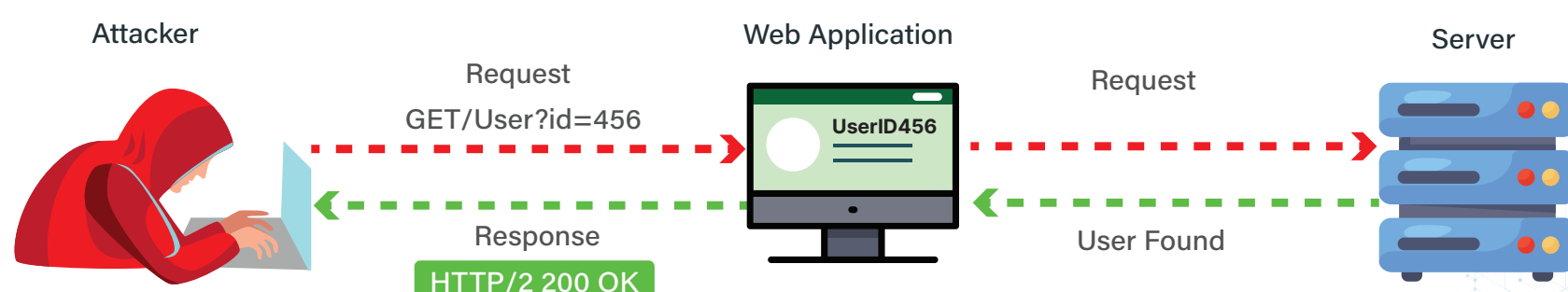
Privilege escalation vulnerability occurs when a user gains unauthorized access to higher levels of privilege than intended. This typically happens due to improper handling of user roles, insufficient input validation, or flaws in access control mechanisms. Attackers exploit these weaknesses to elevate their permissions, gaining control over sensitive functions or data they shouldn't have access to. This can lead to severe security breaches, compromising the integrity, confidentiality, and availability of the application and its data.

3. Types of Privilege Escalation

Vertical Privilege Escalation involves a lower-privileged user gaining higher-level access, such as an ordinary user gaining administrative privileges. This type of escalation allows the user to perform actions and access data that are typically restricted to administrators.



Horizontal Privilege Escalation occurs when a user accesses resources or performs actions reserved for other users with the same privilege level. For example, one user accessing another user's data without authorization, exploiting weaknesses in the application's access controls.





4. Vulnerability that Leads to Privilege Escalation

4.1 Cross-Site Scripting (XSS)

Cross-Site Scripting (XSS) occurs when an attacker injects malicious scripts into content from otherwise trusted websites. These scripts can be executed in the context of another user's session. By exploiting XSS, an attacker can hijack user sessions, gaining access to privileges that the user holds, and perform actions on their behalf.

Example:

```

<script>
  var xhr = new XMLHttpRequest();
  xhr.open("GET", "/admin", true);
  xhr.onreadystatechange = function() {
    if (xhr.readyState == 4 && xhr.status == 200)
      document.write(xhr.responseText);
  };
  xhr.send(); </script>

```

4.2 SQL Injection

SQL Injection is a code injection technique where attackers can execute arbitrary SQL queries by manipulating input data. By exploiting this vulnerability, attackers can bypass authentication mechanisms or escalate their privileges by injecting queries that alter database records, including user roles and permissions.

Example:

```

' OR '1'='1'; UPDATE users SET role='admin' WHERE username='attacker'; --

```

4.3 Insecure Direct Object References (IDOR)

IDOR occurs when an application exposes internal objects to users, allowing attackers to manipulate them directly. Without proper access controls, an attacker can change the references to gain unauthorized access to objects, such as user accounts, leading to privilege escalation.

Example:

```

GET /account?id=123 HTTP/1.1
Host: example.com

// Attacker changes 'id=123' to 'id=1' to access another user's account
GET /account?id=1 HTTP/1.1
Host: example.com

```





4.4 Broken Access Control

Broken Access Control vulnerabilities arise when restrictions on what authenticated users are allowed to do are not properly enforced. Attackers can exploit these flaws to access unauthorized functionalities, often gaining higher-level privileges.

Example:

```
...
POST /change-role HTTP/1.1
Host: example.com
Content-Type: application/json
{
  "username": "attacker",
  "role": "admin"
}
```

4.5 Local File Inclusion (LFI)

Local File Inclusion vulnerabilities occur when an application includes files based on user input without proper validation. Attackers can exploit LFI to read sensitive files or execute arbitrary code, leading to privilege escalation.

Example:

```
...
GET /index.php?page=../../../../etc/passwd HTTP/1.1
Host: example.com
```

4.6 Server-Side Request Forgery (SSRF)

SSRF vulnerabilities occur when an attacker can force a server to make requests to unintended locations, accessing internal services and sensitive data. This can be leveraged to escalate privileges by exploiting internal APIs or services.

Example:

```
...
GET /fetch?url=http://localhost/admin HTTP/1.1
Host: example.com
```

4.7 XML External Entity (XXE)

XXE vulnerabilities occur when XML input containing a reference to an external entity is processed by a weakly configured XML parser. An attacker can exploit this to read arbitrary files on the server or perform SSRF, leading to privilege escalation.

Example:

```
...
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE foo [ <!ELEMENT foo ANY >
<!ENTITY xxe SYSTEM "file:///etc/passwd" >]>
<foo>&xxe;</foo>
```





5. How to Detect Privilege Escalation

5.1 Log Analysis

Regularly scrutinize audit and access logs for anomalies such as unauthorized access attempts or unexpected changes in user roles and permissions.

5.2 Intrusion Detection Systems (IDS)

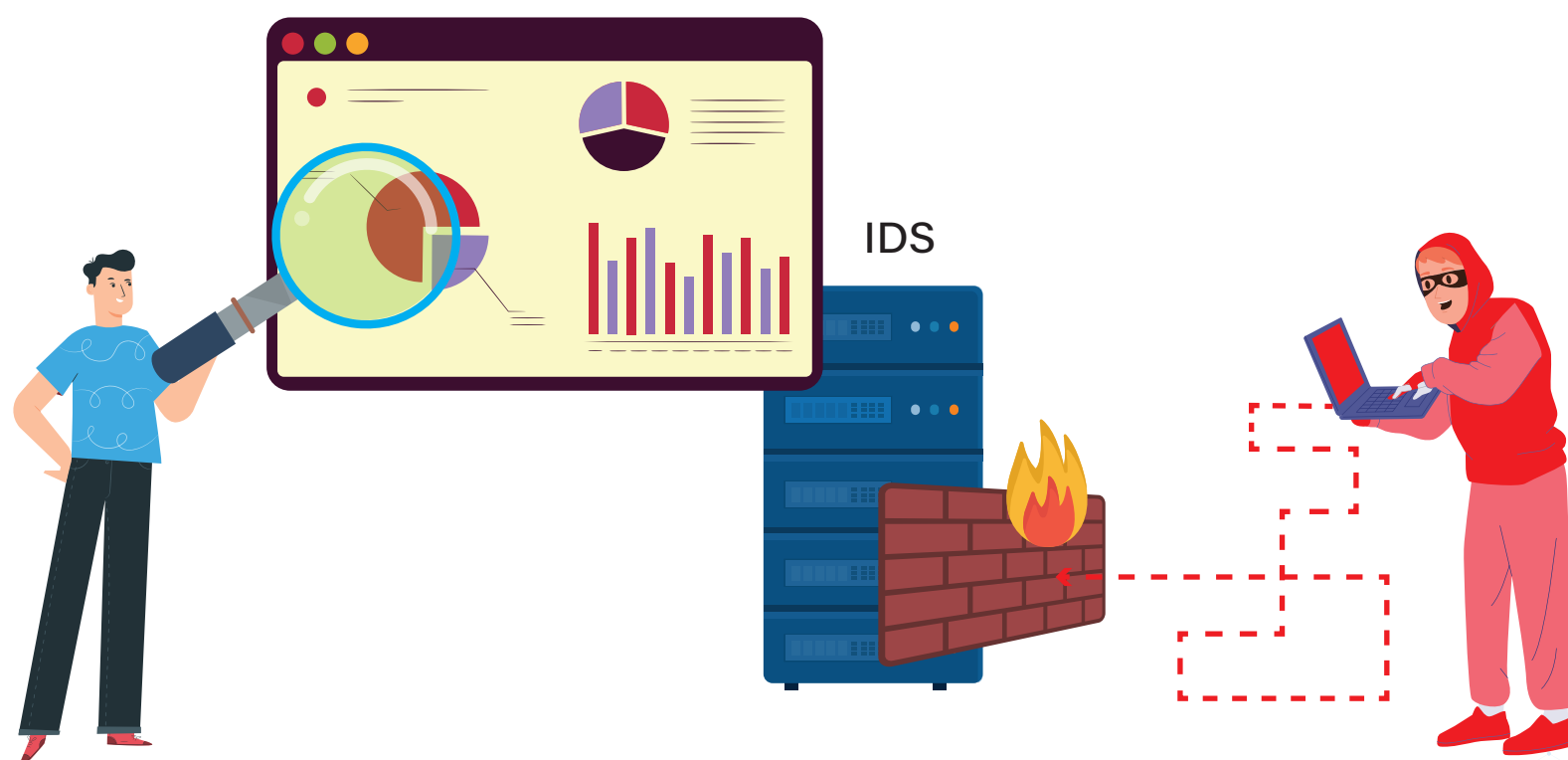
Implement IDS with rule-based and anomaly-based detection capabilities to flag suspicious activities that may indicate privilege escalation attempts.

5.3 Security Information and Event Management (SIEM)

Utilize SIEM platforms to correlate and analyze events across the system, generating alerts for unusual user behaviors or unauthorized access patterns.

5.4 User Behavior Analytics (UBA)

Deploy UBA solutions to establish baseline user behaviors and detect deviations that may indicate unauthorized privilege escalation attempts.





6. Impact of Privilege Escalation

6.1 Unauthorized Access

Attackers exploit vulnerabilities to gain higher privileges than authorized, accessing sensitive data or performing actions reserved for administrators.

6.2 Data Theft and Manipulation

Once escalated, attackers can steal, modify, or delete sensitive data, compromising confidentiality, integrity, and availability.

6.3 Compromise of User Accounts

Privilege escalation can lead to unauthorized access to user accounts, allowing attackers to impersonate legitimate users or perform actions on their behalf.

6.4 Service Disruption

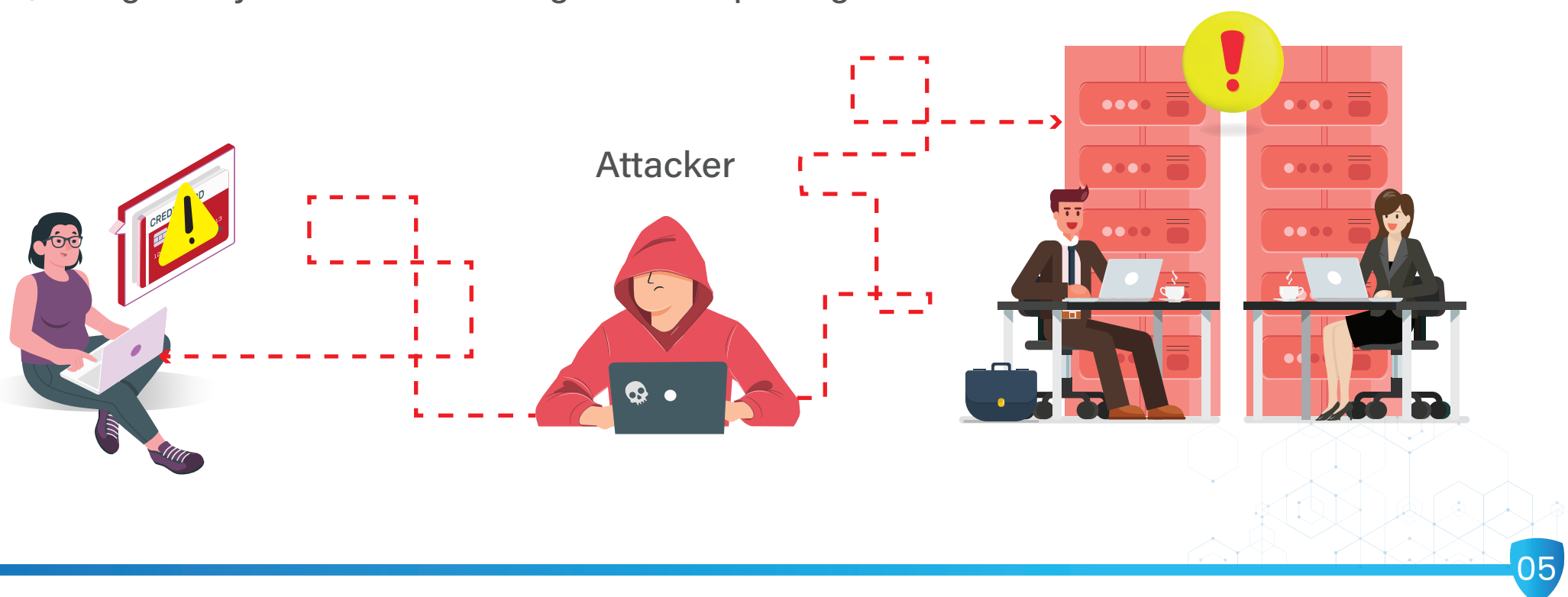
Attackers with escalated privileges may disrupt services, causing downtime, interruptions to business operations, and loss of revenue.

6.5 Reputational Damage

Breaches resulting from privilege escalation undermine trust in the organization's ability to protect data and may damage its reputation with customers, partners, and stakeholders.

6.6 Legal and Regulatory Consequences

Depending on the data compromised and applicable laws, organizations may face legal liabilities, fines, or regulatory sanctions for failing to secure privileged access.





7. Remediation

7.1 Implement Secure Authentication

Utilize robust authentication protocols and mechanisms to secure user access. Store passwords securely using strong hashing algorithms like bcrypt or Argon2 and enforce strict password complexity requirements.

7.2 Apply Principle of Least Privilege (PoLP)

Limit user privileges based on roles and responsibilities. Use tools and frameworks that enforce PoLP across various layers of the application, including database access, API calls, and file system permissions.

7.3 Secure Session Management

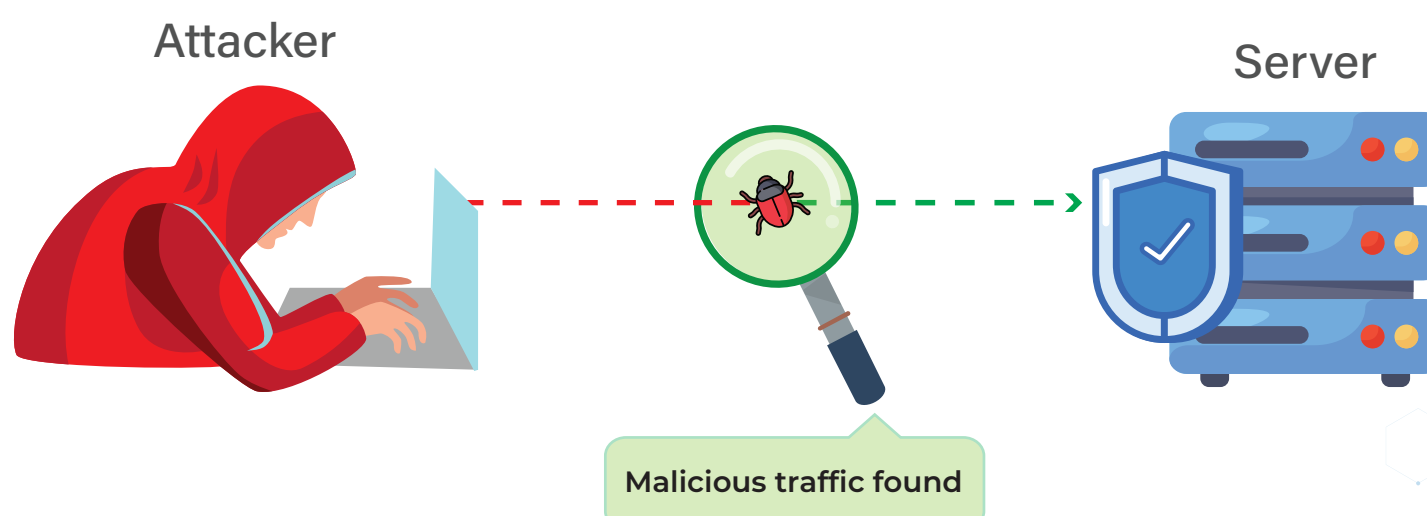
Ensure secure session handling by using HTTPS for all communications, employing secure cookies with HttpOnly and Secure flags, and implementing session expiration mechanisms to minimize the impact of session hijacking.

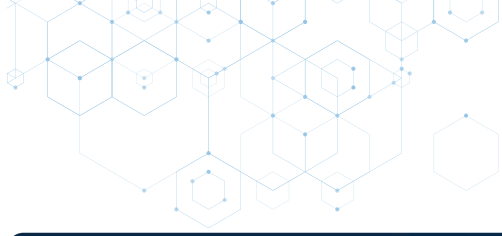
7.4 Employ Defense-in-Depth Strategies

Implement multiple layers of security controls such as network segmentation, firewall rules, and application-level controls like input validation and output encoding. This approach mitigates the risk of privilege escalation through diverse attack vectors.

7.5 Monitor and Control API Access

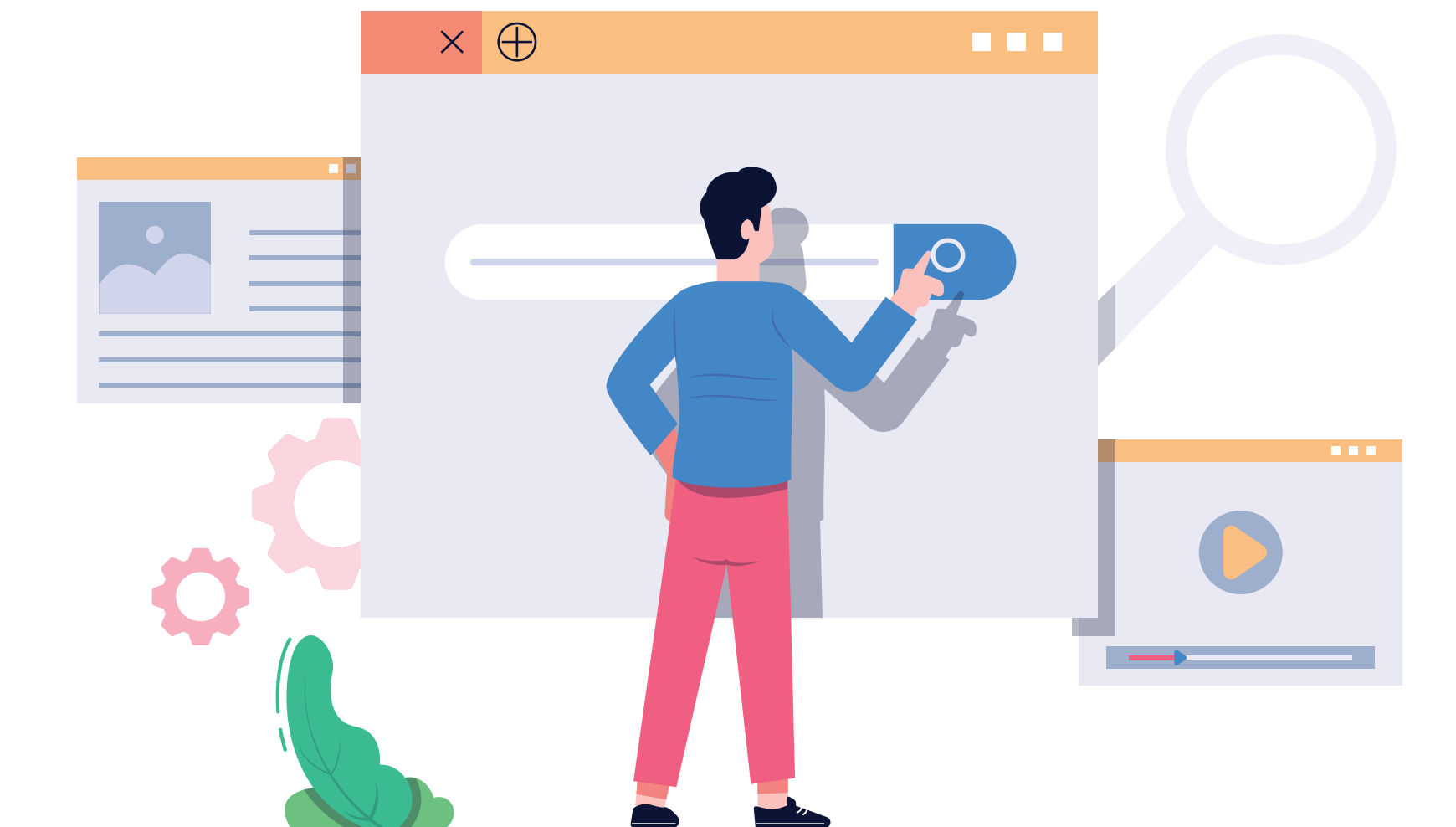
Secure APIs with authentication mechanisms such as tokens or API keys. Implement measures like rate limiting, IP whitelisting, and thorough input validation to prevent unauthorized access and potential privilege escalation.





8. Additional Resources and References

- <https://portswigger.net/web-security/access-control>
- <https://attack.mitre.org/tactics/TA0004/>
- https://owasp.org/www-project-web-security-testing-guide/latest/4-Web_Application_Security_Testing/05-Authorization_Testing/03-Testing_for_Privilege_Escalation
- https://owasp.org/Top10/A01_2021-Broken_Access_Control/
- https://cheatsheetseries.owasp.org/cheatsheets/Authorization_Cheat_Sheet.html

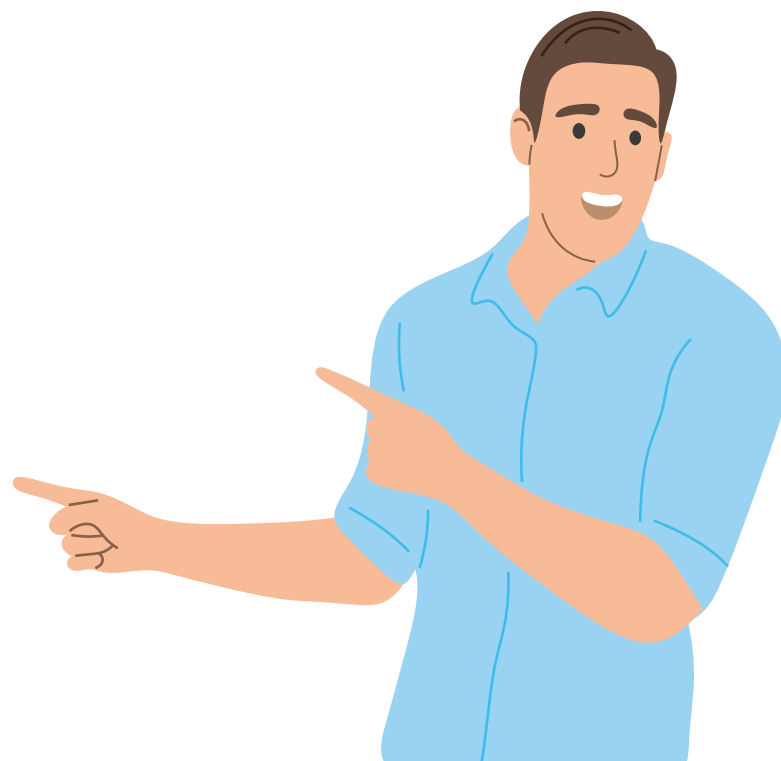




SECURITYBOAT

Frontline Of Your Business

PRIVILEGE ESCALATION HANDBOOK



Scan QR Code to Download Handbook

www.securityboat.net