Cyber Security Fundamentals

Assignment-7

N Ravinder Reddy

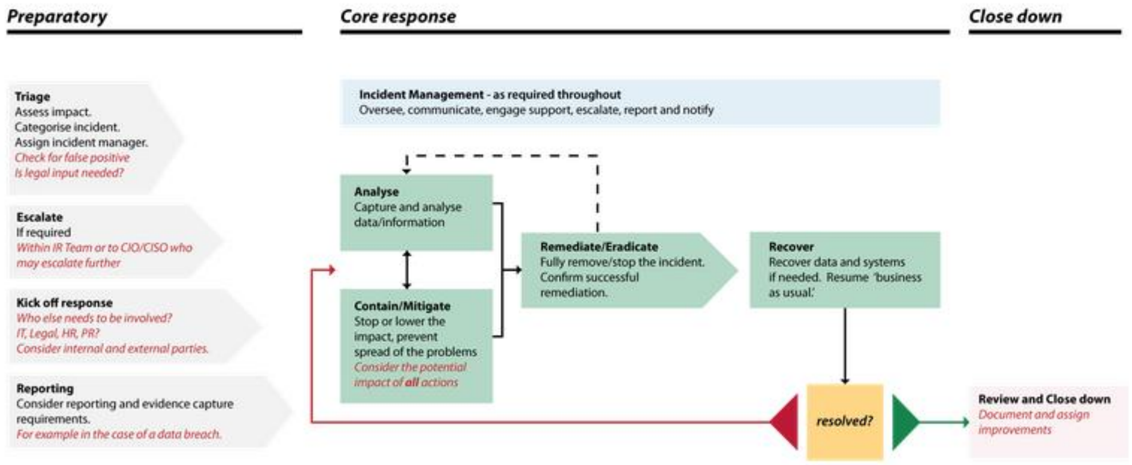
Roll No: 2406CYS106

Q. 1. 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.

Ans: These are the guidelines and standards that your team will follow to handle incidents effectively and consistently. They should cover topics such as incident identification, classification, escalation, notification, containment, analysis, eradication, recovery, documentation, and closure

An incident response team is responsible for handling security-related situations or incidents, investigating and responding to incidents, making command decisions based on the best interests of the business, and ensuring proper data and information handling during incident response and potentially any legal actions resulting from such an incident. The team is composed of a cross-section of various business groups, made up of professionals who come to the rescue when an emergency arises. The team will include technical personnel, management personnel, and legal and communication experts. The team must be ready to respond to an incident the moment it occurs. The response phase of incident response is the point at which the incident response team begins interacting with affected systems and attempts to keep further damage from occurring as a result of the incident.


Preparatory
Core response
Close down
Triage
Assess impact.
Categorise incident.
Assign incident manager.
Check for false positive
Is legal input needed?
Escalate
If required
Within IR Team or to CIO/CISO who may escalate further
Kick off response
Who else needs to be involved?
IT, Legal, HR, PR?
Consider internal and external parties.
Reporting
Consider reporting and evidence capture requirements.
For example in the case of a data breach.
Incident Management - as required throughout
Oversee, communicate, engage support, escalate, report and notify
Analyse
Capture and analyse data/information
Contain/Mitigate
Stop or lower the impact, prevent spread of the problems
Consider the potential impact of all actions
Remediate/Eradicate
Fully remove/stop the incident. Confirm successful remediation.
Recover
Recover data and systems if needed. Resume 'business as usual.'
resolved?
Review and Close down
Document and assign improvements

Categorization of an incident

You should also determine what type of incident you are facing. Some examples include:

- Malicious code: Malware infection on the network, including ransomware
- Denial of Service: Typically a flood of traffic taking down a website, can apply to phone lines, other web facing systems, and in some cases internal systems.
- Phishing: Emails attempting to convince someone to trust a link/attachment.
- Unauthorised Access: Access to systems, accounts, data by an unauthorised person (internal or external) – for example access to someone's emails or account.
- Insider: Malicious or accidental action by an employee causing a security incident.
- Data breach: Lost/stolen devices or hard copy documents, unauthorised access or extraction of data from the network (usually linked with some of the above).
- Targeted attack: An attack specifically targeted at the business - usually by a sophisticated attacker (often encompassing several of the above categories).

Category matrix

As with severity, it is very useful to create a matrix of the different categories.

You can enhance this by adding examples of different severity incidents alongside each category. This will help guide and inform your response.

Escalation - decision-making and authorities

Escalation

Typically, matrices are used to determine the severity or priority of an incident. The severity level will inform how quickly the incident needs to be handled and who it might need to be escalated to.

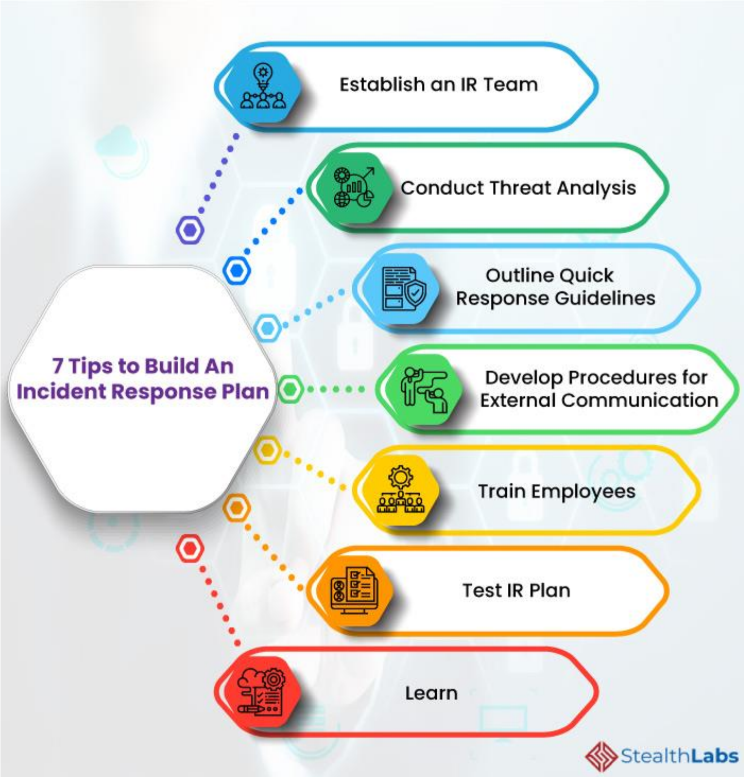
For example, a high or critical severity incident is likely to always need to go up to CIO or board level. A low priority incident could most likely be handled by the IT security team alone. You should document who the escalation points of contact are, along with their contact details (including out of hours) and how quickly the escalation needs to occur.

Authorities

The people escalated to must have the authority to make critical decisions. For example, when a decision may result in major business impact, such as taking a critical service or system offline.

Identify the people who are empowered (or who hold delegated authority) to make such decisions and ensure the escalation process includes these key personnel, as appropriate. It is also important to consider deputies and a process to enable others to make decisions, should the primary contact not be available.

In addition to generic guidance, it may be useful to identify specific situations where the technical team should act autonomously, based on the highest business risks, and where taking early containment action is likely to reduce the impact of particular incidents.


7 Tips to Build An Incident Response Plan
Establish an IR Team
Conduct Threat Analysis
Outline Quick Response Guidelines
Develop Procedures for External Communication
Train Employees
Test IR Plan
Learn
StealthLabs

Incident detection is the process of identifying threats by actively monitoring assets and finding anomalous activity. Once a threat is detected, appropriate actions are taken to neutralize the threat (if it is an active threat at the time of the response) and investigate the incident.

Incident detection is the process of identifying threats by actively monitoring assets and finding anomalous activity (NIST, 2018). Once a threat is detected, appropriate actions are taken to neutralize the threat (if it is an active threat at the time of the response) and investigate the incident. After responding to the incident, the first step in the recovery process is to restore access and availability of systems, networks, services and data to a pre-incident state (NIST, 2018).

Recovery also involves an element of planning that requires the identification, creation, and ultimate implementation of measures for resilience and to enable the restoration of systems, networks, services, and data that were unavailable, harmed, damaged, and/or compromised during the incident. An essential element in ensuring resilience is having an up-to-date *business continuity plan* or *emergency management plan* (Maras, 2014b), which outlines instructions to be followed and actions to be taken in the event of a cybersecurity incident. Put simply, this plan includes detailed information on the ways in which to respond to an incident and recover from it. All those involved in cybersecurity response and recovery should be informed about the emergency managementplan. Here, training is required that includes exercises designed to test the efficacy and efficiency of these plans. An example of this type of exercise is the US Department of Homeland Security's Cyber Storm exercises, which involves participants from national public and private agencies, as well as agencies from other countries (e.g., Australia, Canada, Denmark, Finland, France, Germany, Hungary, Italy, Japan, New Zealand, the Netherlands, Sweden, Switzerland, and the United Kingdom), in order to test current information sharing practices between these agencies, and their cybersecurity preparedness, protection, and response capabilities.

Cybersecurity measures are designed to protect people, property, systems, networks, data, and related resources from threats. These measures include prevention, identification, response to, and recovery from cybersecurity incidents. Assets are identified, and threats, vulnerabilities, and risks are assessed before the development and implementation of cybersecurity measures. Security research can help inform the development of new and enhancement of existing cybersecurity measures. Cybersecurity measures often fail to consider the people who utilize them. Humans can either facilitate or interfere with cybersecurity efforts. For this reason, effective cybersecurity measures and systems need to be created with the individuals who will use them in mind.

Incident response communication plans should address this quandary by outlining clear criteria for when the team should notify law enforcement. The plan should also identify who on the team has the authority to make that determination and what internal notifications should take place prior to involving law enforcement.

The most common communication channels include email alerts, social media, or a dedicated page website or embedded status plugin. Workplace chat tools such as Slack have become incredibly popular for internal communications.

There are five main communication channels for incident communication:

1. A dedicated status page.
2. Embedded status.
3. Email.
4. Workplace chat tool.
5. Social media.
6. SMS.

Having an established communications plan will benefit your organization's ability to handle incidents, while attempting to maintain its reputation, keep its message simple and consistent, and ensure accurate and timely information is released to the appropriate audience.

Reporting and documentation is a critical action that will always occur before, during and after Incident Response. required in keeping with best practices and with the Incident Response plan. The type of reports that might be required might vary but should help in managing and reviewing incidents satisfactorily.

An Incident Response Plan is a written document, formally approved by the senior leadership team, that helps your organization before, during, and after a confirmed or suspected security incident.

A comprehensive incident report is required in keeping with best practices and with the Incident Response plan. The type of reports that might be required might vary but should help in managing and reviewing incidents satisfactorily.

Legal and Regulatory Requirements

No matter the type of business, assuming the organisation is based in, or does business within the UK or EU, GDPR and DPA regulations will need to be adhered to.

Nearly all organisations will hold personal data for employees and for their customers. There may also be specific regulatory requirements for the sector, and potentially, specific customer reporting requirements based on any contractual agreements.

Minimum preparations in this area should include engaging legal advice and documenting:

- What constitutes a reportable incident, based on the types and volumes of data your business holds (including any data held by suppliers)
- When and how to engage legal support
- Extra steps required. For example, preservation of evidence or recording of actions taken

To further improve this, the following should also be considered:

- Create forms ready for any regulatory reporting
- Run workshops focused on scenarios which invoke legal / regulatory requirements and rehearse the appropriate steps

Law enforcement and evidential handling

If the decision is made to undertake any legal proceedings (e.g. prosecute a criminal) then there will also be a requirement to engage relevant law enforcement agencies. This (along with any civil cases) may require careful handling of evidence. ACPO has advice on this process.

Not only does Incident Response Planning prepare you better to face security incidents with confidence, but it also helps your organization mitigate damage to your operations, strengthen relationships with your stakeholders and shareholders, improve your interdepartmental communications, and, eventually, make you.
A well-organized incident response team with a detailed plan can mitigate the potential effects of unplanned events. An incident response plan can speed

up forensic analysis, minimizing the duration of a security event and shortening recovery time.
By having a comprehensive incident response plan in place, your organization will be better prepared to respond quickly and effectively to any cyber incidents that occur. This can help protect against financial losses and reputational damage, ultimately leading to increased customer trust and satisfaction.

Benefits of Incident Response Plan

Here are three of the main benefits of creating an incident response plan for any emergency.

1 Reduce Downtime

One of the main advantages of following an incident response plan is that it will significantly reduce downtime for your company.

A managed service provider will create a detailed action plan for every situation, and give employees guidance on the best way to respond to various incidents.

An IT provider will also create and upload data backups each day to an offsite cloud server. These data backups will give your company the peace of mind to know that your information is well-protected and you can quickly access this data from another location with an internet connection.

2 Maintain Public Trust

Another benefit of using an incident response plan is that it is an excellent way to maintain public trust in the face of an emergency. For example, quickly recovering data from a natural disaster will help the public realize that your company understands the importance of developing a proactive business continuity plan.

On the other hand, the loss of significant data makes it much more difficult to regain the trust of the public and significantly damages the reputation of your company. Investing in an incident response plan is well worth the cost for any company and an IT provider will ensure that your company can quickly bounce back from any situation.

3 Remain in Compliance

Remaining in compliance is critical for many organizations, especially in the healthcare and legal industries. Failure to follow data security protocols can result in substantial fines and costly lawsuits.

Many businesses cannot afford to take any shortcuts and violate these strict regulations. However, the creation of a business continuity plan and incident handling will help ensure that your organization follows all of the rules in your particular industry. An IT service provider will also stay up to date on the latest standards and help your business create a detailed plan for a variety of situations to remain in compliance.

A business continuity plan provided by an IT support company is the most effective way to prepare for any emergency. A managed service provider will also constantly look for ways to improve the business continuity plan to ensure that your company can overcome any situation.

Minimizing downtime, maintaining public trust, and remaining in compliance are just a few of the many advantages of using an IT service company in today's workplace.

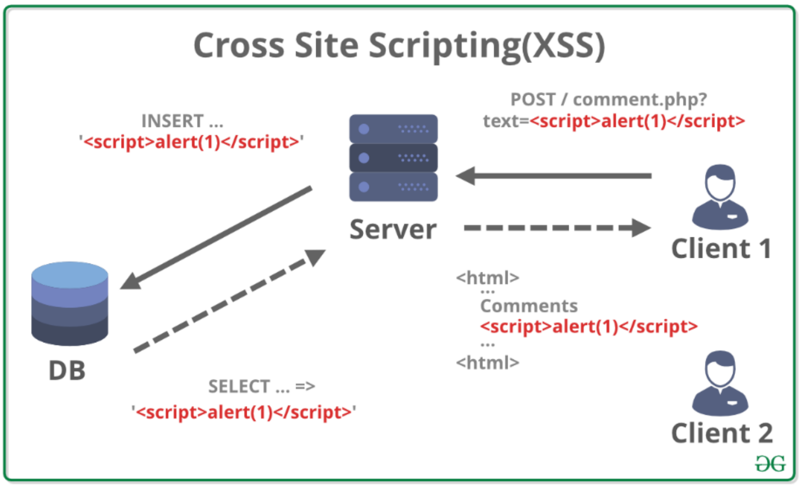
Of course, a cyber attack or natural disaster can happen at any time, but it is the mission of an IT provider to keep your data protected and help your business create a detailed incident response plan.

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

Ans: XSS is a client-side vulnerability that targets other application users, while SQL injection is a server-side vulnerability that targets the application's database.

SQL injection attacks

If you're not yet familiar with SQL (Structured Query Language) injection attacks, or SQLi, here is a great explain-like-I'm-five video on SQLi. You may already know of this attack from xkcd's Little Bobby Tables.


Cross Site Scripting(XSS)
INSERT ...
'<script>alert(1)</script>'
POST / comment.php?
text=<script>alert(1)</script>
Server
Client 1
DB
<html>
...
Comments
<script>alert(1)</script>
...
<html>
SELECT ... =>
'<script>alert(1)</script>'
Client 2

Essentially, malicious actors may be able to send SQL commands that affect your application through some input on your site, like a search box that pulls results from your database. Sites coded in PHP can be especially susceptible to these, and a successful SQL attack can be devastating for software that relies on a database (as in, your Users table is now a pot of petunias).

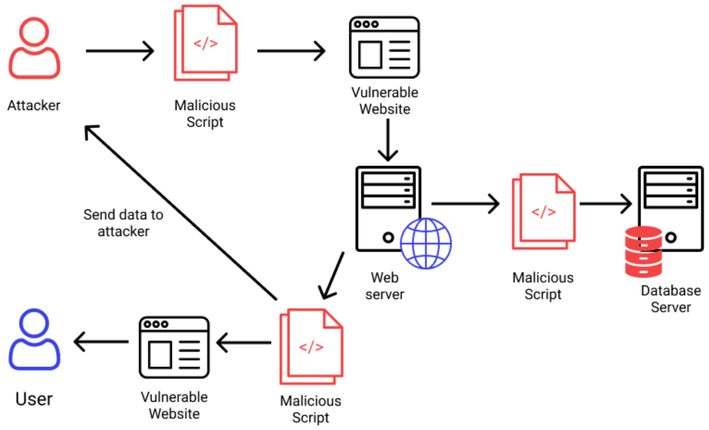
SQL injection mitigation

In order to effectively mitigate SQL injections, developers must prevent users from being able to successfully submit raw SQL commands to any part of the site.

Some frameworks will do most of the heavy lifting for you. For example, Django implements the concept of Object-Relational Mapping, or ORM, with its use of QuerySets. We can think of these as wrapper functions that help your application query the database using pre-defined methods that avoid the use of raw SQL.

Being able to use a framework, however, is never a guarantee. When dealing directly with a database, there are other methods we can use to safely abstract our SQL queries from user input, though they vary in efficacy. These are, by order of most to least preferred, and with links to relevant examples:

1. Prepared statements with variable binding (or parameterized queries),
2. Stored procedures; and
3. Whitelisting or escaping user input.



If you want to implement the above techniques, the linked cheatsheets are a great starting point for digging deeper. Suffice to say, the use of these techniques to obtain data instead of using raw SQL queries helps to minimize the chances that SQL will be processed by any part of your application that takes input from users, thus mitigating SQL injection attacks.

Cross-site scripting attacks, also called XSS attacks, are a type of injection attack that injects malicious code into otherwise safe websites. An attacker will use a flaw in a target web application to send some kind of malicious code, most commonly client-side JavaScript, to an end user. Rather than targeting the application's host itself, XSS attacks generally target the application's users directly. Organizations and companies running web applications can leave the door open for XSS attacks if they display content from users or untrusted sources without proper escaping or validation.

XSS vulnerabilities are one of the OWASP Top 10 security concerns today, especially as so many organizations rely heavily on web applications for customer interaction and validation. However, by writing secure code, testing for vulnerabilities, and working with security tools like Veracode Dynamic Analysis, developers can prevent, detect, and repair potential vulnerabilities allowing for XSS exploitation.

What is Cross Site Scripting (XSS)?

XSS occurs when an attacker tricks a web application into sending data in a form that a user's browser can execute. Most commonly, this is a combination of HTML and XSS provided by the attacker, but XSS can also be used to deliver malicious downloads, plugins, or media content. An attacker is able to trick a web application this way when the web application permits data from an untrusted source — such as data entered in a form by users or passed to an API endpoint by client software — to be displayed to users without being properly escaped.

Because XSS can allow untrusted users to execute code in the browser of trusted users and access some types of data, such as session cookies, an XSS vulnerability may allow an attacker to take data from users and dynamically include it in web pages and take control of a site or an application if an administrative or a privileged user is targeted.

Malicious content delivered through XSS may be displayed instantly or every time a page is loaded or a specific event is performed. XSS attacks aim to target the users of a web application, and they may be particularly effective because they appear within a trusted site.

XSS attack mitigation

In all of these cases, XSS attacks can be mitigated with two key strategies: validating form fields, and avoiding the direct injection of user input on the web page.

Validating form fields

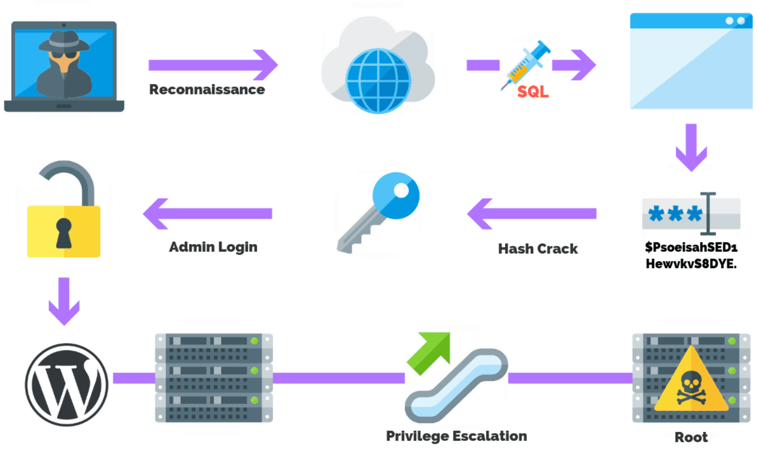
Frameworks can again help us out when it comes to making sure that user-submitted forms are on the up-and-up. One example is Django's built-in Field classes, which provide fields that validate to some commonly used types and also specify sane defaults. Django's EmailField, for instance, uses a set of rules to determine if the input provided is a valid email. If the submitted string has characters in it that are not typically present in email addresses, or if it doesn't imitate the common format of an email address, then Django won't consider the field valid and the form will not be submitted.

If relying on a framework isn't an option, we can implement our own input validation. This can be accomplished with a few different techniques, including type conversion, for example, ensuring that a number is of type int(); checking minimum and maximum range values for numbers and lengths for strings; using a pre-defined array of choices that avoids arbitrary input, for example, months of the year; and checking data against strict regular expressions.

Thankfully, we needn't start from scratch. Open source resources are available to help, such as the OWASP Validation Regex Repository, which provides patterns to match against for some common forms of data. Many programming languages offer validation libraries specific to their syntax, and we can find plenty of these on GitHub. Additionally, the XSS Filter Evasion Cheat Sheet has a couple suggestions for test payloads we can use to test our existing applications.

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

Ans: Privilege escalation refers to a network attack aiming to gain unauthorized higher-level access within a security system. It typically starts with attackers exploiting vulnerabilities to access a system with limited privileges.
There are two main types of privilege escalation: horizontal and vertical. You need to understand these types of privilege escalation and how to protect against privilege escalation in general.



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.

Not every attack will provide threat actors with full access to the targeted system. In these cases, a privilege escalation is required to achieve the desired outcome. There are two types of privilege escalation attacks including vertical and horizontal.

Vertical Privilege Escalation

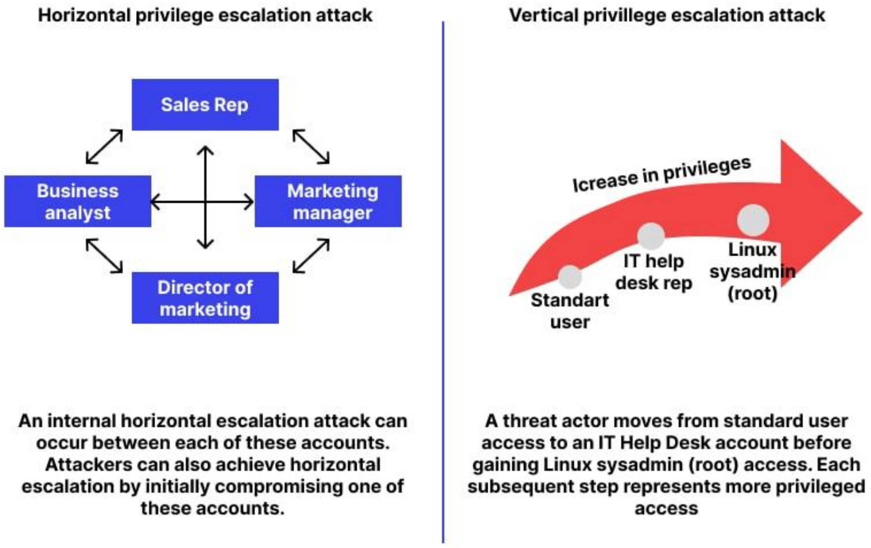
Vertical privilege escalation occurs when an attacker gains access directly to an account with the intent to perform actions as that person. This type of attack is easier to pull off since there is no desire to elevate permissions. The goal here is to access an account to further spread an attack or access data the user has permissions to.

Day in and day out I analyze numerous phishing emails that attempt to perform this attack. Whether it’s a “bank”, “Amazon”, or any other countless number of ecommerce sites, the attack is the same. “Your account will be deactivated due to inactivity. Please click this link and login to keep your account active.” This is, however, one example of many cookie-cutter phishing templates seen in “the wild”.

Horizontal Privilege Escalation

Horizontal privilege escalation is a bit tricky to pull off as it requires the attacker to gain access to the account credentials as well as elevating the permissions. This type of attack tends to require a deep understanding of the vulnerabilities that affect certain operating systems or the use of hacking tools.

Phishing campaigns have been used to perform the first part of the attack to gain access to the account. When it comes to elevating permissions, the attacker has a few options to choose from. One option is to exploit vulnerabilities in the operating system to gain system or root-level access. The next option would be to use hacking tools, like Metasploit, to make the job a bit easier.


Horizontal privilege escalation attack
Sales Rep
Business analyst
Marketing manager
Director of marketing
An internal horizontal escalation attack can occur between each of these accounts. Attackers can also achieve horizontal escalation by initially compromising one of these accounts.
Vertical privillege escalation attack
Icrease in privileges
Standart user
IT help desk rep
Linux sysadmin (root)
A threat actor moves from standard user access to an IT Help Desk account before gaining Linux sysadmin (root) access. Each subsequent step represents more privileged access

Examples Of Privilege Escalation Attacks

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 Passwd 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).

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.

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.

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.

Preventing privilege escalation attacks requires a multifaceted approach that incorporates various security practices, tools, and measures. Here are best practices to consider:

1. Carefully manage privileged accounts

Here are several ways to adequately manage access and prevent privilege escalation:

- Limiting the number of privileged accounts: Reduce the number of privileged accounts to the minimum necessary, and only grant elevated permissions to users who require them.
- Least privilege principle: Assign the minimum level of access and permissions required for users to perform their tasks.
- Regularly reviewing and auditing: Periodically review and audit privileged account permissions to ensure that only authorized users have elevated access.
- Monitoring and logging: Implement monitoring and logging of privileged account activities to detect suspicious actions and potential abuse.
- Use of temporary credentials: Implement the use of temporary or time-limited credentials for privileged accounts, which expire after a certain period.

2. Patch and update software

Regularly patching and updating software, operating systems, and firmware is essential to address known vulnerabilities and reduce the risk of privilege escalation attacks. Develop a patch management process that includes:

- Monitoring for updates: Keep track of security patches and updates released by software vendors. Use automation of tools or processes wherever possible.
- Prioritizing patches: Prioritize patches based on the severity of vulnerabilities and the potential impact on your systems.
- Testing and deployment: Test patches in a controlled environment before deploying them to production systems to avoid potential compatibility issues or disruptions.
- SCA and SAST: Don't forget software composition analysis and static application security testing. If you are using third-party web tools, make sure your libraries are up to date with the Dev team.
- Change tickets: Notify your Security Operations team when you make specific changes that may affect their function or vision across the organization.

3. Perform Vulnerability Scans

Regular vulnerability scanning helps identify potential weaknesses, misconfigurations, and vulnerabilities in your systems that could be exploited in a privilege escalation attack. Implement a vulnerability management program that includes:

- Regular scanning: Schedule vulnerability scans to run regularly on your systems and networks. This includes, as stated above, regular application security testing for vulnerabilities and known exploits in Developer tools (e.g. JexBoss, Apache Struts).
- Remediation: Establish a process for prioritizing and remediating identified vulnerabilities, based on their severity and potential impact. This can also include upgrading from LDAP to LDAPS, and NTLM to Kerberos wherever possible.
- Validation: Verify that vulnerabilities have been successfully remediated and that new vulnerabilities have not been introduced during the process.

4. Monitor Network Traffic and Behavior

Monitoring network traffic and user behavior can help detect potential privilege escalation attacks in progress or identify signs of unauthorized access. Implement network and behavior monitoring solutions, such as:

- Intrusion detection systems (IDS): Use IDS to monitor network traffic for signs of intrusion or malicious activity.
- Security information and event management (SIEM): Employ SIEM tools to collect, analyze, and correlate log data from various sources, helping to identify potential security incidents.
- User and entity behavior analytics (UEBA): Implement UEBA solutions to monitor and analyze user behavior for signs of unusual or suspicious activity that may indicate unauthorized access or privilege escalation attempts.

5. Enforce a Strong Password Policy

A strong password policy reduces the risk of unauthorized access and privilege escalation through password attacks. Ensure your password policy includes:

- Complexity: Require passwords to include a mix of upper and lowercase letters, numbers, and special characters.
- Length: Enforce a minimum password length, typically 12-16 characters.
- Password rotation: Set a password expiration period, requiring users to change their passwords regularly. Make sure they cannot repeat passwords for at least 3 cycles.
- Account lockout: Implement account lockout policies to lock accounts after a specified number of failed login attempts, reducing the risk of brute-force attacks.

6. Conduct Security Awareness Training

Educate your employees about the risks of privilege escalation attacks and the importance of following security best practices. Security awareness training should cover:

- Recognizing social engineering tactics: Train employees to identify and respond to phishing attempts, pretexting, and other social engineering tactics that could be used to gain unauthorized access.
- Safe password practices: Educate users about the importance of creating strong, unique passwords and never sharing them with others.
- Reporting suspicious activity: Encourage employees to report any unusual or suspicious activity they encounter, such as unexpected privilege changes or unauthorized access attempts.
- Following organizational security policies: Ensure that employees understand and adhere to your organization’s security policies and procedures, including those related to access control, software updates, and privileged account management.
- Regular training and updates: Conduct security awareness training on a regular basis and keep employees informed about emerging threats, vulnerabilities, and best practices to stay protected.

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

Ans: 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.

- Brute-force attack.
- Keylogger attack.
- Dictionary attack.
- Credential stuffing.
- Man-in-the-middle.
- Traffic interception.
- Phishing.
- Password spraying.

A typical password cracking attack looks like this:

- Get the password hashes.
- Prepare the hashes for a selected cracking tool.
- Choose a cracking methodology.
- Run the cracking tool.
- Evaluate the results.
- If needed, tweak the attack.
- Go to Step 2.

BRUTE FORCE ATTACK

In a brute-force password attack, a hacker tries to access a secure user account through trial and error. This typically involves systematically entering every possible combination of letters, numbers, and symbols into a password field until one works.

A brute force attack uses trial-and-error to guess login info, encryption keys, or find a hidden web page. Hackers work through all possible combinations hoping to guess correctly.

These attacks are done by ‘brute force’ meaning they use excessive forceful attempts to try and ‘force’ their way into your private account(s).

This is an old attack method, but it's still effective and popular with hackers. Because depending on the length and complexity of the password, cracking it can take anywhere from a few seconds to many years.

It is a serious illegal and unethical crime that can result in severe legal consequences. The risk of password cracking can cause significant harm to individuals and organizations, including data theft, financial loss, and damage to reputation.