

Information Security Research

Delgerekh Ariunbileg

November 19, 2025

1 Topics

1.1 Describe 3 different ways by which data leakage/exfiltration can occur in an enterprise network.

Data leakage can occur when a Trojan horse (e.g. SilentBanker) is propagated into a network by concealing itself as legitimate software. Internal users within an enterprise's network may download software that they believe is legitimate, when in reality, it allows bad actors to gain unauthorized access. Trojan horses can intercept sensitive data before it is encrypted, acting as a man-in-the-browser. As a result, bad actors may retain keystroke logs, read files, or forward sensitive information to their remote computers.

Secondly, lack of access control and assigned privilege roles may give internal folks, within an enterprise, the ability to access and alter data that they are not authorized or permitted to do so. For example, an employee may be given full control capabilities when they should only be able to read a file [9]. Bad actors may exploit this vulnerability to access sensitive data, which they can use to exfiltrate or alter.

Lastly, another cause of data leakage is the theft of enterprise owned assets such as laptops and cell phones. Employees working from a public location compromise data when they leave their devices in unsecured spaces such as cafes and airports. Bad actors could make copies of the data or steal the devices to remove the hard disk drive [1] before the victim reports the crime.

1.2 Identify 3 controls (preventive, detective, and compensating) that could be applied to prevent and detect leakage.

As a preventative measure, organizations should implement the principle of permitting least privilege. Damage is minimized when internal stakeholders are only permitted the access that they need to perform their duties. Privilege may be granted depending on the conditions that the enterprise sets up. For example, an organization may choose to use a cryptographic key in addition to user authentication, which can enable a user to be given access to additional systems.

A detective method that can be adopted includes the integration of an intrusion detection system (IDS), which monitors users and their activities, scans for vulnerabilities and misconfigurations, and assesses the current system to ultimately raise alarms real time for enterprises. IDSs are critical for all enterprises, but especially those that handle sensitive data and therefore require accountability, or currently handle complex network environments.

A compensating method that could aid enterprises recover from disruptions and/or failures would be to keep back-up repositories of their sensitive data. High performing enterprises should take efforts to use on premise or cloud based solutions to back-up or safekeep their information if an emergency occurs. This method would allow the enterprise to prevail and continue operations until the vulnerabilities that caused the failure are patched [4].

1.3 Identify remediation steps an organization can apply to respond to a data leakage event.

To remediate damages, enterprises may follow a similar approach to ensure they appropriately respond to a data leakage event. These steps should be laid out in their incidence response plan:

- Reduce additional exposure by containing the leakage

- Block the attack using an intrusion prevention system (IPS) that terminates the session.
 - IDS can redirect traffic to a monitoring host, adjust performance to slow the attack, or shut down the network particularly/entirely.
 - Enterprise systems should revoke all privilege to access data if a user's credentials were compromised in the first place.
- Monitor and collect data to understand breach
 - Monitor the network and collect data from the time of the breach to understand the causation.
 - Pull various audit logs and visualize the information to determine the data that was accessed and how access was gained.
 - Ultimately, the enterprise may want to know who carried out the attack.
 - Keep any evidence that may gather enough information to bring the attackers to justice.
- Address weaknesses and enhance future security
 - Determine the cause of the incident and address the needs that must be taken.
 - Improve and refine incident response plan to improve key performance indicators such as mean time to repair, mean time between failures, and more.
 - Educate and train employees to spur strong security practices.
 - High performing organizations spend \$8,000 per employee on training [4].

1.4 What type of Operating Systems do you think are more secure – Open source or proprietary?

Open source operating systems offer more security than proprietary operating systems. This is attributable to the community driven evaluations and focus, which allow for vulnerabilities to be identified and patched without the constraint of financial resources. In addition, the open design allows for organizations to tailor to their cryptographic needs over time and configure source code to strengthen security measures.

Open source code can be evaluated by a community of experts who are passionate about scanning and fixing errors. Community led efforts are more likely to attract true experts, rather than salaried employees who are tasked with finding errors.

Unlike proprietary operating systems, open source code also receives extensive public scrutiny, which is vastly different from security with obscurity. Open design allows for a transparent path to security, unlike proprietary which can mask vulnerabilities.

Proprietary code may only be prioritized when pertaining to specific vulnerabilities, and financial resources can be a point of constraint for pushing patches. Open source code is continually reviewed and patched through community efforts. On a related note, open source vulnerabilities may be fixed over the weekend due to community efforts, and not postponed until the workweek.

Open source cryptographic algorithms allow for users or organizations to adopt cryptography based on their own needs. For instance, SandboxAQ provides a library of cryptographic algorithms that can be tailored and applied to defend all three elements of the CIA triad when protecting data [13].

Users of open source OS can also scale their security needs over time. They can modify source code and its configurations to tailor to their high security needs [6].

1.5 Identify and briefly describe one specific software vulnerability that may exist in a software application utilized by an organization and the potential financial, operational, health & safety impact, if exploited.

Buffer overflows can exist in a software application, where more data than can be handled is given to the allocated memory storage, causing the excess data to “overflow” to another memory location. If exploited, buffer overflows allow malicious actors to enter the user's program code, system data,

or system program code through the injection of malicious programs, which can lead to disruptions [10].

For example, a malicious actor could interact with a hospital's appointment organizer, that has no limit set for the buffer, to insert ransomware into another memory location. Consequently, the ransomware could encrypt system data that allows employees at a hospital to carry out operations, such as other users' names, health records, insurance information, etc. This could disrupt normal operations until the ransomware is paid, placing a strain on an organization's finances and delaying critical patient care [3].

1.6 Identify one (a) preventative, (b) detective, (c) corrective, (d) recovery, (e) deterrence, and (f) compensating control an organization should implement to reduce the likelihood that impact is realized by the organization.

- **Preventative measure**

- A preventative measure is limiting the allocated buffer space and user input, which disbars users from exploiting buffers in the first place.

- **Detective control**

- A detective control could include the monitoring of the changes made in the database or any unauthorized processing running on the machine.
 - This detective control would allow organizations to continually review and recognize potential patterns/behaviors that can lead to impact.

- **Corrective measure**

- A corrective measure could be to isolate any backed-up databases while the affected systems are continually analyzed.
 - By isolating their back-ups, firms can ensure that these databases are protected from similar exploits.

- **Recovery method**

- As a recovery method, the organization could establish and utilize data from the backed-up system that is verified to be clean and unaffected.
 - After isolation and close analysis of backed-up data quality, the organization can patch the vulnerabilities that were exposed and introduce the back-up as the primary database.

- **Deterrence measure**

- A deterrence measure could include a user warning directed to the consequences of unauthorized access, such as punishment by law.
 - This is a natural deterrent that could potentially discourage an attack, although unlikely if the actor is already an accomplished attacker.

- **Compensating control**

- Lastly, a compensating control could be the implementation of a data execution prevention (DEP) system that segments the memory to not execute when a buffer overflow occurs.
 - Therefore, making the executable malicious code obsolete from the stack (most common type of buffer overflow) [?].

- [1] Amazon AWS. "SSD Vs HDD - Difference Between Data Storage Devices - AWS." Amazon Web Services, Inc, aws.amazon.com/compare/the-difference-between-ssd-hard-drive/.
- [2] AnuPriya. "Alleged Data Breach Targets Indonesian Educational Platform Via SQL Injection." Cyber Security News, 13 Mar. 2025, cyberpress.org/data-breach-indonesian-educational-platform/.
- [3] Blackduck. "The Heartbleed Bug." Heartbleed Bug, www.heartbleed.com/.
- [4] CMU SEI. "Cyber Intelligence Tradecraft Report." Software Engineering Institute, www.sei.cmu.edu/documents/1589/2019_011_001_546699.pdf.
- [5] Codacy. "Insecure Design: A Complete Guide." Codacy | Blog - Automate Your Code Quality, blog.codacy.com/insecure-design-owasp-top-10.
- [6] James, Parker W. "What Are the Advantages of Using an Open-source Operating System?" 9 July 2025, www.cyberly.org/en/what-are-the-advantages-of-using-an-open-source-operating-system.
- [7] Lenaerts-Bergmans, Bart. "What Is an Injection Attack? | CrowdStrike." CrowdStrike: We Stop Breaches with AI-native Cybersecurity, www.crowdstrike.com/en-us/cybersecurity-101/cyberattacks/injection-attack/.
- [8] Microsoft Learn. "Data Execution Prevention." Microsoft Learn: Build Skills That Open Doors in Your Career, learn.microsoft.com/en-us/windows/win32/memory/data-execution-prevention.
- [9] ---. "Determine Permission Levels and Groups in SharePoint Server." Microsoft Learn: Build Skills That Open Doors in Your Career, learn.microsoft.com/en-us/sharepoint/sites/determine-permission-levels-and-groups-in-sharepoint-server.
- [10] Neumetric. "Buffer Overflow Attacks: Understanding, Mitigating, and Preventing." Neumetric, 13 Oct. 2023, www.neumetric.com/buffer-overflow-attacks.
- [11] OWASP. "WSTG - Latest." OWASP Foundation, the Open Source Foundation for Application Security | OWASP Foundation, owasp.org/www-project-web-security-testing-guide/latest/4-Web_Application_Security_Testing/07-Input_Validation_on_Testing/05-Testing_for_SQL_Injection.
- [12] ---. "A04 Insecure Design - OWASP Top 10:2021." OWASP Foundation, the Open Source Foundation for Application Security | OWASP Foundation, owasp.org/Top10/A04_2021-Insecure_Design.
- [13] SandBoxAQ. "Open "Sandwich" Speeds Crypto Agility." Transforming the World with AI and Advanced Computing | SandboxAQ, 1 Oct. 2025,

[www.sandboxaq.com/press/sandboxaq-launches-sandwich-an-open-source-meta-library-of-cryptographic-algorithms
to-speed-the-implementation-of-agile-cryptography.](http://www.sandboxaq.com/press/sandboxaq-launches-sandwich-an-open-source-meta-library-of-cryptographic-algorithms-to-speed-the-implementation-of-agile-cryptography)

[14] SQLMAP. "Introduction." Sqlmap: Automatic SQL Injection and Database Takeover Tool, sqlmap.org/.