SEVEN BASIC SECURITY CHECKS FOR EVALUATING EDUCATIONAL PLATFORMS

Good security—keeping data safe from unauthorized access or use and unintended or inappropriate disclosure—is integral to good privacy. Educational technology companies are required under various laws to maintain "reasonable" security programs. In addition, signatories of the **Student Privacy Pledge** have committed to provide a program that is "reasonably designed to protect the security, privacy, confidentiality, and integrity of student personal information." How should parents, teachers, and administrators evaluate whether a company's security practices are reasonable?

Educators, Administrators and Parents:

The following are a few simple steps that anyone can follow to determine whether or not an educational platform's Web and mobile applications have implemented basic security safeguards. While these steps will not guarantee that an app or platform has incorporated all possible security measures, **they provide a basic threshold** for evaluating whether reasonable security standards have been met. No two companies or products are the same and each must be considered within the context for which they were designed, and in which they are actually used by the school or district.

This document is not intended to mean that every company who follows all these steps is sufficiently secure, or that a company who does not is insecure. When in doubt, always follow up by asking the company directly about their security safeguards and practices. This is not a de facto security test for Pledge signatories.

1. Look for an Encrypted Connection Where Appropriate (Usually)

Properly implemented encryption converts information into an unintelligible form that can only be decoded (or "decrypted") by the intended recipient. Encryption is a fundamental component of securing data while in transit and while it is "at rest" (stored on a local server or "in the cloud"). Students' personal information should be routinely encrypted in transit and at rest. Encryption is fundamentally important for any pages that contain sensitive data, or that collect usernames, passwords, payment details, or other personally identifiable information. This test is not foolproof for detecting whether data is encrypted in transit, but provides a reliable indicator of how encryption is supported by a service.

- If there is an exchange of student data (PII) via the website, make sure the URL in the browser starts with **https://**. If you attempt to load the page via http:// (by removing the "s"), *it should force you back to https*.
- Repeat after logging in. See if you can still access the site.
- Try this on multiple pages within the website, especially pages where you're being asked to supply personal information.
- **Caution**: If PII is not collected or maintained, encryption may not be necessary. Further, the collection or exchange of PII may happen through some other method that

would not put at risk data shared through an unsecure user browser, in which case again this type of encryption in transit identified by use of https:// may not be necessary.

2. Ensure That Applications Use TLS Between Email Servers

Transport Layer Security (TLS) is a protocol – a standardized way for internet services to "talk" to each other – that provides communication security between client/server applications that communicate with each other over the Internet. It enables privacy, integrity (more info here) and protection for the data that's transmitted between different nodes on the Internet. You may also see the term "SSL." TLS is a successor to the secure socket layer (SSL) protocol.

- **One method to see if a company uses TLS is to look up the company's domain name ("*examplecompany.com*") in Google's Safer Email Transparency Report.** This will enable you to see how much of the ed tech site's email exchanged via Gmail is encrypted in transit. The Report posts daily percentages of email sent and received using TLS for the sites with the most traffic to and from Google's mail servers—so if a company sends enough email traffic to Gmail users, it will show up here.
- **Caution:** Smaller companies may not be listed in Google's Safer Email Transparency Report. If the company is not reflected, you should be able to contact the company directly and ask about their security measures, including TLS.

3. Ensure That URLs Do Not Contain Sensitive Information

Personal information should never appear in clear-text within a URL. If a web-based platform includes sensitive or personal information in URLs, they are putting that information at risk of unintended disclosure.

- Ensure the URL at login does not display any clear text information
    - URLs have occasionally been found to reflect the username, password, date of birth, social security number, or other PII. For example: "https://www.foogames.com/platform/John-Smith/1995/classroom-12" reflects the user's name, birth year, and group name within the school.
    - A URL may contains a numeric identifier that doesn't directly reveal user information but can be easily edited to find other users within the same group. For example: "https://www.educationgames.com/platform/3398571/7-mec/landscape" - could be edited to change the last numeral to "2" (3398572), potentially providing access to another student's account page.
- After navigating a web-based platform, review your web browsing history and its cache to ensure that URLs you have visited do not contain sensitive information.
- Test sites that use unique identifiers in URLs – by changing a number and refreshing – this will show whether the site has basic authorization and validation controls in place.
- **Caution:** This is not automatically a problem just because there is a formatted number in the URL; many sites have appropriate validation and authorization measures in place.

4. Ensure Sensitive Information is Not Stored in the Cache or Browser History

Browsers improve performance by storing copies of web pages that a user views, so that they can be displayed again later without the need to re-fetch them from the network ("caching"). Although caching improves browser performance and provides efficiency to users, caching of pages that contain sensitive information risks exposing that information to unauthorized access.

If a computer is a shared resource, is available in a public forum, or is simply left unattended without being secured, the next user can see where the previous person has been, and possibly gain access to their personal information. Ensure that sensitive information is not being cached by taking the following steps (more detail here):

- Log in to a web-based service and browse to a page with sensitive information (e.g. a class roster viewed from a teacher's account, or the page that asks you to submit personal information to set up an account).
- Sign out of the service, or fill in the requested information and then leave the site.
- Click "Back," "undo close tab," or otherwise return to the page. If the original page reappears, *with user information still displayed*, this indicates the page was stored in the browser cache with sensitive information intact, not appropriately protected.
- **Caution:** Because of standard http protocols allowing retention in history/storage, there is no way for an education platform to *completely* control this behavior. Many browsers honor server cache controls via the "back" button, but they are not required to.

5. Ensure That Passwords Are Protected

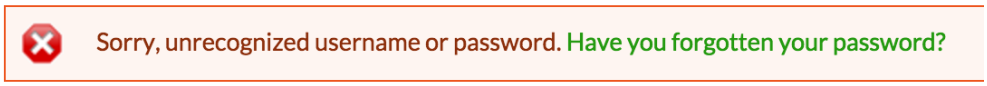
Passwords are an imperfect access protection, but they remain the most common, and there are ways to ensure that they are implemented in as effective a way as possible.

- Check that the login page is appropriately protected**.** Perhaps the most important protection for passwords is safeguarding them as they are sent from the user at login. Refer to Checks #1-2 above to ensure a secure connection is established at any point where you are asked to enter a password.
- Check that the platform handles lost or forgotten passwords appropriately**.** In the process for recovering lost passwords, be aware of the following security risks:
    - The platform should not send you a new password via email. Email has a long life and may sometimes be transported without encryption. It's better to send an https:// password reset link**.** Ideally, the link should expire after one use or after a short time if not used.
    - The platform should not send or display your current password**.** If an application is able to show or send your current password in plain text, it indicates that it has been *stored* in plain text or another recoverable format.
        - If the site does not send you a password in clear text when you say you "forgot your password," that is probably sufficient. However, if you want additional reassurance, you can always ask the company what process they use.
        - The best practice for password storage is for the platform to perform a one-way hash (a form of encryption) and store the hashed result. This requires the passwords to be "cracked" in the event of a security breach. Although hashing must be done correctly, it can provide an extra layer of security for passwords obtained via breach or inadvertent release.
    - **Caution:** Security questions do not necessarily solve these problems**.** While they may not hurt, security questions do not necessarily add much protection for password security. Security questions often rely on fixed information that can be guessed or easily obtained, or that never changes, and as a result, they often do not provide reliable, consistent protection for end users.

6. Ensure that the Login and Password Recovery Mechanisms Do Not Reveal Unnecessary Information (e.g. the Existence of an Account)

These kinds of vulnerabilities do not pose an enormous risk to most users, but they can be useful for an attacker who wants to target a specific user account, such as the administrator. Follow these steps to check for good security at the user-login and password-recovery stages:

- Try to log in using both an invalid username and an invalid password; then using a valid username with an invalid password. In both cases, the application should provide identical error responses. If a site provides different responses, it can reveal whether the username itself was valid. The following is an example of a good response—it does not reveal any unnecessary information, but merely states that *either* the username *or* password was incorrect (without identifying which one caused the error).



  In contrast, messages like the ones below are less secure, because they reveal the validity of the username used in the login attempt.

- Similarly, try requesting a new password using the "lost or forgotten password" recovery function—using both a valid and an invalid username. Check whether the application gives *different* responses for valid and invalid usernames. A good response will not reveal the validity of the username. For example, an application should respond by something like: *"if an account with that username exists, an email has been sent."*
- **Caution**: This is not an absolute; many companies strategically choose not to obscure this information between the two levels (login ID and password) because of sufficient additional safeguards. However, it is worth checking as some less sophisticated companies have been known to allow even admin password resets via email, which means the site could be compromised quickly and with little to no technical skill.

7. Be Watchful for "Information Leakage"

Information leakage occurs when an application reveals or "leaks" sensitive information about the system or its users. There is no standardized method to test for information leakage—rather, users should go through a full range of interactions with the application and be watchful for extra information being revealed in the platform's responses and transactions. Common ways that information can be leaked include:

- Password-recovery mechanism reveals a user's email account (if the username is not the user's email address) or a user's registered name;
- Teacher account requests for class roster information might return sensitive information beyond what is necessary about each of the students in a class;
- General site use via the student or parent account reveals information about other users.

These safeguards are most applicable only if and where the platform maintains personally identifiable information. These technological safeguards should be carried out in conjunction with physical, administrative and other protocols. Many additional evaluations can (and should) be done by security

experts. Parents and administrators should also remember that cyber threats come from other sources, such as email phishing and malware.

For more information, we recommend:

- For general information about on-line security, an excellent baseline is the OWASP standard. https://www.owasp.org/index.php/Main_Page
- For education specific security information, Common Sense Media *Information Security Primer for Evaluating Educational Software* provides detailed introduction to testing. https://www.commonsense.org/education/privacy/security-primer