Vanderbilt REDCap Security Information

Last updated 11/2015

I. Environment

The VICTR environment in which the Vanderbilt instance of REDCap is installed is implemented with known best practices and is consistent with guidance from organizational security documentation. The team supporting the applications are trained in the secure operation of Linux environments and configure the systems to meet standards of least privilege, and least function. REDCap uses encryption mechanisms to protect traffic between the Web server and the End User. Encryption is also used to protect REDCap authenticators.

The Vanderbilt REDCap operating environment undergoes a vulnerability scan at least annually. The REDCap source code is continually assessed in the change management process, and also by automated tools by members of the REDCap Consortium. Results of automated scans are submitted to the REDCap team to validate findings and remediate any vulnerabilities as needed.

VICTR systems are backed up regularly by the groups that hosts the application environment. Critical systems are identified and adequate backup procedures are in place to ensure that all necessary information is backed up.

The Environment Disaster Recovery Plan is included in the Contingency Plan.

The Emergency Mode Operation Plan is addressed in Vanderbilt Emergency Procedures and in the Contingency Plan for the hosting environment.

End users are responsible for protecting the workstation that they use to connect to the environment and for taking reasonable precautions such as installing malware detection/prevention mechanisms, using strong passwords, encrypting hard drives or other media types used, and enabling local firewall policies.

Stored PHI in REDCap is stored in the protected datacenter facility. Encryption is not used due to the physical protection controls that are in place for the environment.

II. Software

Risk Management is built into the System Development Life Cycle (SDLC) for VICTR applications. Security measures are identified and implemented as needed to mitigate risks.

REDCap employs various methods to protect against malicious users who may attempt to identify and exploit any security vulnerabilities in the system. All incoming data gets intentionally filtered, sanitized, and escaped. This includes all data submitted in an HTTP Post request and all query string data found in every URL while accessing REDCap, among other modes through which user-defined data gets submitted in the application.

Server environment variables that are vulnerable to forgery by users are also checked and sanitized. All user-submitted data is properly filtered for any possibly harmful markup tags (e.g. <script>) and is then escaped before ever being displayed on a web page within the application. SQL queries sent to the database server from REDCap are all properly escaped before being sent. If any values used in an SQL query originated from

user-defined values, they would have already been sanitized beforehand as well, as described above. User-defined data used within SQL queries also have their data type checked to prevent any mismatching of data types (e.g. making sure a number is really a number). These processes of sanitization, filtering, data type checking, and escaping all help to protect against methods of attack, such as Cross-Site Scripting (XSS) and SQL Injection. To specifically protect against Cross-Site Request Forgery (CSRF), which is another method of attack, REDCap utilizes a "nonce" (a secret, user-specific token) on every web form used in the application. The nonce is generated anew on each web page as the user navigates within REDCap during a session.

In addition, REDCap has a feature to defend against denial of service attacks so that after it detects a high number of access attempts within a short period of time, the offending IP address will be blocked from accessing the site.

Many institutions that have installed REDCap have made use of enterprise-level web application security scanners, such as HP Webinspect and Acunetix, to scan and test REDCap's security and its ability to withstand various methods of attack. REDCap has performed very in such instances.

The REDCap source code is evaluated continuously for security and functional flaws. The software development lifecycle involves weekly bug/security fix releases and monthly functional change releases.

The REDCap Team supports the code base that is distributed to the REDCap consortium. Updates for the REDCap software are released weekly. These updates may include items specific to security.

To ensure that REDCap users have access only to data and information that they are supposed to have within the application, user privileges are utilized within the software. Each user has their own account, and their user account will only have access to REDCap projects that they themselves have created or to projects which other users have granted them access.

User privileges are also granular on the project level and can be modified within any given project by someone with proper privileges accessing the project's User Rights page. The creator of a project will automatically be given full rights to everything within the project, after which they may grant other users access to the project and limit their user privileges as desired. Within each project, there are user controls to limit access to various functionality and modules, such as being able to export data, to enter data, to add or modify database fields or survey questions, to build or run reports, to modify user privileges, to view the logging records, and so on. Another feature called Data Access Groups can be implemented to help segregate users and the data they enter by placing users into data access groups, after which they will only be able to access records created by someone in their group. This particular feature is entirely optional but is especially helpful in certain situations, such as for multi-institutional projects where the data entered by one institution should not be accessible or viewable by other institutions with access to that same project.

III. REDCap Authentication REDCap uses LDAP authentication for most users. Where a user cannot be assigned a Vanderbilt account, table-based authentication is used.

Table based password policy: Passwords must match the same hardening criteria as for the Vanderbilt Password Policy, as well as password lifetime maximum and minimum settings. REDCap contains customizable settings that govern login activity, including an option to manually set the number of failed login attempts before a user is locked out of the system for a specified amount of time. In the Vanderbilt instance of REDCap, these settings are

5 failed login attempts within a 15-minute period.

REDCap utilizes two-factor authentication. When connecting to the Vanderbilt instance of REDCap from outside of the Vanderbilt VPN, users must login using a username/password combination and an authentication code that is sent via email, SMS, or Google authenticator mobile app to the information in the user's REDCap account. Two-factor login authentication can be disabled on the project level, at the Project Owner's request.

REDCap contains an auto-logout setting, which is tomizable (default auto-logout time is 30 minutes), and will automatically log a user out of the system if they have not had any activity (e.g. typing, moving the mouse) on their current web page for the set amount of time.

This prevents someone else from accessing their account and their project data if they leave a workstation without properly logging out or closing their browser window.

IV. User Management

Project Owners are responsible for initiating new user accounts, authorizing users to access their projects, and configuring the correct user permissions within each project. Project Owners are also responsible for correctly denoting PII as such so that the user permissions are correctly applied.

In VICTR applications, the Project Owner or designee is responsible for authorizing user access to the PHI contained in their REDCap projects and to ensure that they have appropriate clearance.

In REDCap, once a user is no longer authorized to access PHI, the user can be permanently deleted from the system or can put in suspension status. Suspending a user allows them to remain a user in the system but restricts access to any projects in REDCap until their status is reactivated. For users needing access to a project for a set period of time, an expiration date can be set on the account to automatically revoke permissions. The termination of API's for REDCap is linked to the user account that requested the token. Once the users account is restricted, the API is restricted. Suspending a user, setting a user's expiration date, or deleting a user from the REDCap environment must be done through the REDCap administrative team. It is the Project Owner's responsibility to contact the administrative team for any such user updates.

The Project Owner or designee are responsible for setting up access and modifying access rights as needed.

REDCap has the ability to create User Roles at the project level so that each project can implement role- based user management procedures to easily ensure that users are granted the proper access.

The User Management dashboard is provided in REDCap to assist Project Owners/administrators in managing the users in their respective projects. In the dashboard access to a project can easily be enabled or disabled.

V. Audit

In VICTR applications, the Project Owner or designee is responsible for auditing for suspicious or unauthorized activities. Auditing capabilities are included in VICTR applications for this purpose.

REDCap also provides a monthly reminder to Project Owners/administrators to update the personnel who have access to their respective projects. The dashboard lists the users who have access to each project, the last time they logged in, and allows access to be granted or taken away. It is the Project Owner's responsibility to authorize access to their data and to update that access as appropriate.

REDCap maintains a built-in audit trail that logs all user activity and all pages viewed by every user, including contextual information (e.g. the project or record being accessed). Whether the activity be entering data, exporting data, modifying a field, running a report, or add/modifying a user, among a plethora of other activities, REDCap logs all actions. The logging record can itself be viewed within a project by users that have been given privileges to view the Logging page. The Logging page allows such users to view or export the entire audit trail for that project, and also to filter the audit trail in various ways based upon the type of activity and/or user. The built-in audit trail in REDCap allows administrators to be able to determine all the activity and all the data viewed or modified by any given user.

REDCap logs ALL access to data. If PHI were altered by an unauthorized personnel, it would be reflected in the history and audit trail.