

Softwaresicherheit

Eine Präsentation von
Benedikt Streitwieser und Max Göttl

Einführung Kryptographie und IT-Sicherheit

Gliederung

- Einleitung: Was ist Softwaresicherheit
- Populäre Beispiele
- Anforderungen der Softwaresicherheit
- Gefahren
- Konkrete Beispiele:
 - SQL Injection
 - Buffer Overflow
 - Cross Site Scripting

Einleitung: Was ist Softwaresicherheit?

- Beschäftigt sich mit der Sicherheit von Programmen
- Schutz vor Gefahren bzw. Bedrohungen
- Vermeidung von wirtschaftlichen Schäden und Minimierung von Risiken
- Schwachstellen in jedem noch so gut geplanten und umgesetzten System
- Bedrohung der Grundprinzipien der Informationssicherheit
- Angriffe auf die Schutzziele bedeuten für Unternehmen Angriffe auf reale Unternehmenswerte, im Regelfall das Abgreifen oder Verändern von unternehmensinternen Informationen

Populäre Beispiele

- Stagefrigth
- Shellshock
- Heartbleed
- Explosion of Ariane 5
- Pentium Processor division Error
- Patriot-Missile Error

Anforderungen der Softwaresicherheit

- **Verfügbarkeit** (Availability): Das Programm darf nicht abstürzen und muss immer wieder in einen Zustand zurückkehren in dem neue Eingaben verarbeitet werden können.
- **Vertraulichkeit** (Confidentiality): Geheime Informationen, wie z.B. Passwörter dürfen nicht in öffentlich lesbaren Ausgaben/Speicherbereichen auftauchen
- **Integrität** (Integrity): Öffentliche Eingaben dürfen den Inhalt bestimmter Speicherbereiche sowie das Verhalten des Programms nicht beeinflussen.

Gefahren

- SQL Injection
- Buffer Overflow
- Cross Site Scripting
- DOS Exploits
- Improper Error Handling
- ...

SQL Injection

- Ausnutzen einer Sicherheitslücke in Zusammenhang mit SQL-Datenbanken
- Entsteht durch mangelnde Maskierung oder Überprüfung von Metazeichen in Benutzereingaben
- Angreifer versucht dabei, über die Anwendung, die den Zugriff auf die Datenbank bereitstellt, eigene Datenbankbefehle einzuschleusen

SQL Injection

Ziele:

- Ausspähen, Manipulation und Löschen von Daten
- Datenbank-Server verändern
- Einschleusen von beliebigem Code
- Erlangen von Administrationsrechten

SQL Injection

Angriffsarten:

- Blind SQL Injection:
 - Content-based Blind SQL Injection
 - Time-based Blind SQL Injection
- SQL Injection

SQL Injection

Beispiel:

- Datenbank enthält Informationen zu Studenten
- Abfrage mittels Matrikelnummer: MatrNr:
- SQL Statement: `SELECT * FROM student WHERE matrNr = 1234;`
- Resultat: Details des Studenten mit Matrikelnummer 1234

SQL Injection

Abwehrmaßnahmen:

- Input Validation, z.B. mit regular expression
- Whitelisting
- Blacklisting
- Prepared Statements
- DB Zugriffsberechtigungen:
 - Read only
 - Zugriff nur auf benötigte Datenbanken und Tabellen

SQL Injection

- Möglicher Angriff:
- Böartiger Code wird eingegeben MatrNr:
- Folgen:
 - Mit Abwehrmaßnahme (Prepared Statement):
 - Details des Studenten mit Matrikelnummer 1234 werden angezeigt
 - Böartiger Code hat keine Auswirkungen
 - Ohne Abwehrmaßnahmen:
 - Schädliche Eingabe wird ausgeführt
 - Informationen aller Studenten werden angezeigt

Buffer Overflow

- Gehört zu den häufigsten Sicherheitslücken in aktueller Software
- Fehler im Programm
- Speicher wird überschrieben weil nicht genügend reservierter Speicher vorhanden ist
- Angezeigte Fehler sind beispielsweise „Programm.exe funktioniert nicht mehr“ (Windows) oder „Segmentation fault“ (Linux)

Buffer Overflow

- Absturz des betroffenen Programms
- Verfälschung von Daten
- Injizieren von Code
- Angreifer erlangt Zugang zu System

Buffer Overflow

Beim Start eines Programms, weist ihm das Betriebssystem Speicherbereich zu

- In einem Teil liegt der geschützte und nicht veränderbare Programmcode
- Darüber liegt der **Heap** , in dem das System globale Variablen und Konstanten ablegt
- Dann folgt der **Stack** , der lokale Variablen und den Inhalt von Prozessorregistern aufnehmen kann

Buffer Overflow

- Im Stack liegen auch die Rücksprung-Adressen von Unterprogrammen
- Beim Buffer-Overflow wird eine lokale Variable mit mehr Inhalt gefüllt, als für Sie reserviert ist
- Der Trick der Hacker besteht jetzt darin, die Rücksprungadresse auf Programmsegmente zu lenken, die den eigentlichen Schadcode enthalten.

Buffer Overflow

Abwehrmaßnahmen:

- Zahl der zu schreibenden Zeichen begrenzen
- Beispielsweise Java statt C oder C++ verwenden
- StackShield: sichert die Return-Adresse und korrigiert sie bei Bedarf
- StackGuard: versucht die Rücksprungadressen zu schützen
- Nicht-ausführbarer Stack
- Prüfen der Abbruchbedingung in Schleifen
- moderne Compiler nutzen
- Dem Programmhersteller vertrauen
- Den besten Schutz für den PC gewährleisten auf jeden Fall regelmäßige Updates
- Alternative Software

Cross Site Scripting

- Ausnutzen einer Sicherheitslücke in Webanwendungen
- Informationen werden in einen Kontext eingefügt, in dem sie als vertrauenswürdig eingestuft werden
- Aus diesem vertrauenswürdigen Kontext kann dann ein Angriff gestartet werden
- Ziel ist es meist, an sensible Daten des Benutzers zu gelangen, um beispielsweise seine Benutzerkonten zu übernehmen (Identitätsdiebstahl).

Cross Site Scripting

- HTML Injection
- Übergabe von Parametern an ein serverseitiges Skript, das eine dynamische Website erzeugt
- Angriffsarten:
 - Reflektiert (Non-Persistent)
 - Persistent
 - DOM-basiert

Cross Site Scripting

- Folgen:
 - Entführen von Benutzer-Sessions
 - Website-Defacements
 - Einstellen negativer Inhalte
 - Phishing-Angriffe
 - Übernahme der Kontrolle des Benutzerbrowsers

Cross Site Scripting

Beispiel:

- In ein Gästebuch wird schadhafter Code eingefügt:
`<script>alert ("This site has been hacked") ;</script>`
- Der Eintrag wird für alle Besucher angezeigt
- Beim Öffnen des Gästebuches erscheint im Browser ein Fenster mit dem Hinweis „This site has been hacked“

Hier können Sie Ihren eigenen Gästebucheintrag hinzufügen:

Name

Max Mustermann

Eintrag

Test<script>alert ("This site has been hacked");</script>

Absenden

Mein Gästebuch

Am Mi 01.06.2016 um 01:19:40 PM MESZ

schrieb Benedikt
folgenden Eintrag:
Ein kleiner Test

Am Mi 01.06.2016 um 01:33:05 PM MESZ

schrieb Max
folgenden Eintrag:
Halli Hallo

Am Mi 01.06.2016 um 03:30:40 PM MESZ

schrieb Max Mustermann
folgenden Eintrag:
Test

Meldung von Webseite



This site has been hacked

OK

Cross Site Scripting

Abwehrmaßnahmen:

- Input Validation, z.B. mit regular expression
- Ersetzung von Metazeichen
- PHP: Funktionen wie `strip_tags()`, `htmlspecialchars()`, `htmlentities()` verwenden
- Content Security Policy: Webadministrator legt vertrauliche Domains von JavaScript fest
- Benutzer:
 - Unterstützung für JavaScript im Browser deaktivieren
 - Browser aktuell halten