16.5 Sicherheitshinweise

Neben den großen Sicherheitsthemen (Benutzereingaben, XSS, SQL Injection, Passwörter, Übertragung usw.) gibt es noch eine unendliche Liste an Sicherheitsempfehlungen. Grundsätzlich aber gilt die Formel: ***Denkfaulheit == Sicherheitsrisiko***

Hier einige Impulse zur PHP und Websicherheit:

- **Unnötige Technologien abschalten!**
  Man sollte alle Technologien die nicht unmittelbar für die Website benötigt werden deaktivieren. z. B. CGI, Perl, FastCGI.
- **Dateirechte vergeben!**
  Mit chmod kann auf einem Linux-Server jede Datei allein durch die Dateirechte weiter gesichert werden. Ein File, das nur gelesen wird sollte auch keine Schreibrechte haben. Dateirechte lassen sich auch im FTP Client nach- oder während dem Upload festlegen.
- **Intelligente Serveradministration!**
  Die Konfiguration des Webserver ist maßgeblich für die Sicherheit der Website. Von der Firewall über den DNS-Server bis zu den Port Öffnungen – Webserveradministration (ob nun Apache/Linux oder IIS/Windows) ist eine eigene Wissenschaft. Da kann der PHP Code noch so gut geschützt sein, wenn ein Angriff über SSH oder FTP auf den Server erfolgt.
- **Zertifikate verwenden!**
  SSL/TLS Protokolle und Zertifikate steuern das verschlüsselte Übertragen der Daten. Mit ihnen ändert sich http:// zu https:// - also einer sicheren Übertragung. Das **Lets Encrypt** Zertifikat ist gratis – man kann aber auch Zertifikate kaufen.
- **Fehler und Warnungen in der PROD abschalten!**
  PHP Fehlermeldungen und Warnungen geben Hacker 'nützliche' Informationen für ihre 'Arbeit'. Fehlerprotokolle sollten also dementsprechend gut geschützt werden.
- **Saubere Programmierung!**
  Variablen die nicht mehr benötigt werden, sollten eigentlich mit `unset()` wieder gelöscht werden. Jeder Handler (DB-Handler, File-Handler usw.) sollte nach seiner Arbeit auch wieder geschlossen werden. Sessionvariablen und Cookies sollten mit einem vernünftigen Ablaufdatum versehen werden. `E_PARSE` Fehler dürfen niemals im Code sein!
- **Verzeichnisse schützen!**
  Ordner mit Hilfsdateien (z. B. für CSS, Bilder) sollten mit einem Redirect in einer index.php weitergeleitet werden, damit die Inhalte des Ordners nicht mit dem Browser ausgelesen werden. z. B. `header("Location: http://www.meineSeite.at");`
- **Scriptsicherheit erhöhen mit php.ini!**
  In der Konfigurationsdatei php.ini gibt es zahlreiche Optionen welche die Scriptsicherheit erhöhen. z. B. Deaktivieren der globalen Verfügbarkeit von Variablen, Shell-Befehle deaktivieren, Funktionen deaktivieren uvm.
- **Website testen!**
  Am besten sollte eine externe Person die Website testen. Dabei kann auch die Usability erhoben werden. Oft entsteht ein Sicherheitsrisiko in Fragen, die dem PHP-Entwickler sowieso selbstverständlich bzw. irrsinnig sind, und sie diese dann auch nicht stellen.
  "Betriebsblindheit" ist ebenfalls ein Sicherheitsrisiko.